Multiplicative Latent Force Models

Daniel J. Tait

Doctor of Philosophy
University of Edinburgh
2019

Declaration

I declare that this thesis was composed by myself and that the work contained therein is my own, except where explicitly stated otherwise in the text.

(*Daniel J. Tait*)

To my parents

Abstract

Latent force models (LFM) are a class of flexible models of dynamic systems, combining a simple mechanistic model with the flexibility of an additive inhomogeneous Gaussian process (GP) forcing term. These hybrid models achieve the dual goal of being flexible enough to be broadly applied, even for complex dynamic systems where a full mechanistic model may be hard to motivate, but by also encoding relevant properties of dynamic systems they are better able to model the underlying dynamics and so demonstrate superior generalisation. In this thesis, we consider an extension of this framework which keeps the same general form, a linear ordinary differential equation with time-varying behaviour arising from a set of smooth GPs, but now we allow for multiplicative interactions between the state variables and the GP terms. The result is a semi-parametric modelling framework that allows for the embedding of rich topological structure.

Following a brief review of the latent force model, which we note is a particular case of the GP regression model, we introduce our extension with multiplicative interactions which we refer to as the *multiplicative latent force model* (MLFM). We demonstrate that this class of models allows for the possibility of strong geometric constraints on the pathwise trajectories. This will enable the modelling of systems for which the GP trajectories of the LFM are unsatisfactory.

Unfortunately, and as a direct consequence of the strong geometric constraints we have introduced, it is no longer straightforward to carry out inference in these models; therefore the remainder of this thesis is primarily devoted to constructing two methods for carrying out approximate inference for this class of models. The first is referred to as the Bayesian adaptive gradient matching method, and the second is a novel construction based on the method of successive approximations; a theoretical construct used in the standard classical existence and uniqueness theorems for ODEs.

After introducing these methods, we demonstrate their accuracy on simulated data, which also allows for an investigation into the regimes in which each of the respective methods can be expected to perform well. Finally, we demonstrate the utility of the MLFM on motion capture data and show that, by using the framework developed in this thesis to allow for the sharing of a smaller number of latent forces between distinct trajectories with specific geometric constraints, we can achieve superior predictive performance than by the modelling of a single trajectory.

Lay summary

The modelling of complex dynamical systems with time-dependent behaviour is a challenging problem. The *mechanistic philosophy* of model building requires specifying a complete summary of the complex interactions between the components of the system, and how they interact with one another, and such a description is often hard to motivate. Conversely, modern methods in statistics and machine learning prefer to specify a very flexible model, applicable across a diverse range of scenarios, and then allow for the interactions to be inferred by the observed data, and this is often referred to as the *data driven paradigm.* While a wholly specified model is often hard to motivate, they will typically perform better than data-driven methods at predicting the outcome of new observations, especially when the data is sparse relative to model complexity, because they better embody significant physical constraints, and relevant features of the governing dynamics. Hybrid models are an attempt to combine the physical realism of mechanistic models with the flexibility of more general methods, and so achieve a synthesis of the classical systems studied in science and mechanics with those systems that can most easily be injected with a data sensitive probabilistic component.

Latent force models (LFM) are a particular example of such a hybrid model. They are constructed by combining perhaps the most straightforward class of mechanistic models with a flexible Gaussian process (GP) term — the latent forces. GPs are a very flexible class of models for statistical inference of continuous data which allow one to place a measure of uncertainty over large families of unknown functions. However, a dynamical system is more than just some function with a temporal argument, and the LFM framework provides the necessary structure to begin to make these flexible models more realistic by combining these GP terms with the simplest models of classical mechanics. We offer a brief review of the LFM and note that in fact, the LFM is a particular instance of a GP so that existing methods of inference and model fitting can immediately be applied to this model.

However, in part because of their flexibility, GPs are not always appropriate for modelling data that has a high degree of geometric structure, for example data constrained to the surface of a sphere as in the rotation of a fixed length vector around the origin. Moreover, since the LFM is a particular instance of a GP, there are many dynamical systems, which we know a priori will possess a complicated geometry and so will not be adequately modelled by the usual LFM. Therefore in this thesis, we introduce an extension of this model classes which will allow for stronger geometric constraints to be imposed on the model. Mathematically this is achieved by allowing the latent forces to interact multiplicatively with the system state variables in the equation describing how this process evolves, and so we refer to this model as the *multiplicative latent force model* (MLFM).

Unfortunately, the greater control over the geometry of samples from our model comes at the expense of the mathematical tractability of the usual LFM. Therefore we introduce two methods of approximating the MLFM which lead to more straightforward

methods of inference. The first method is an adaptation of existing methods that work by interpolating the observed data and then finding the parameter value which gives the best match between the estimated gradient and the functional form of the evolutionary dynamics. The second is a novel method which uses a set of increasingly accurate approximations to the trajectory to learn the most likely parameters to have produced this trajectory. After introducing these approximations, and describing how they may be used to carry out inference we compare the accuracy of both methods by way of a simulation study.

A heuristic interpretation of the motivation behind the LFM is given by considering a marionette, in this instance, an articulated system with a fixed geometry can be manipulated to display a wide range of different motions by the variations of a relatively small number of strings. In this interpretation, we can view the mechanical component of the LFM as representing the physical structure of the system, and the small number of latent forces as the set of controls which achieve different motions. We demonstrate the validity of this interpretation by applying our MLFM to human motion capture data of a subject performing a golf swing. This is a complex nonlinear process for which a complete physical description is unavailable. Nevertheless, we can demonstrate that the MLFM successfully models such a system and that the learned latent forces have qualitative features that agree with our physical intuition as well as displaying good predictive performance. We conclude the thesis with a discussion and suggest possible areas of future research.

Acknowledgements

First and foremost my most heartfelt appreciation to my supervisor Dr. Bruce Worton, who has shown me an unfaltering level of patience and support during this process. I honestly could not have completed this process without him, nor imagine a more supportive advisor. My sincerest thanks also to Professor Colin Aitken, who has been unfailingly kind and encouraging.

Furthermore I would like to extend my most sincerest thanks to my external examiner Professor Dirk Husmeier and internal examiner Dr Miguel de Carvalho. Their close examination, discerning questions and helpful feedback greatly improved the quality of this work.

For so many reasons I cannot begin to thank my parents enough for their endless support and patience; before, during and no doubt after this project.

This project was funded by the Engineering and Physical Sciences Research Council. The data used in this project was obtained from `mocap.cs.cmu.edu`. The database was created with funding from NSF EIA-0196217.

Contents

Chapter 1

Introduction

Historically the modelling of dynamical systems has broadly followed one of two distinct philosophies; the first of these is classical and often referred to as the *mechanistic approach* which aims to construct realistic models guided by sound scientific principles. In contrast, the *data driven* paradigm, inspired by modern machine learning techniques, places a greater emphasis on prediction and allowing the observables to guide the process of pattern discovery. The conflict between these two philosophies can be particularly pronounced for complex dynamic systems when a complete mechanistic description is often difficult to motivate, but models with some degree of physical realism are likely to be more effective extrapolating from the training data. Therefore it would be desirable to have a framework that allows for the specification of a simple representation of the driving dynamics, while still allowing for relevant dynamic systems properties to be encoded into the models.

In this chapter, we provide a brief review of dynamical systems and some of the statistical methods used to model them with a particular focus on continuous time, smooth, dynamical systems described by differential operators, which we regard as those systems for which there is the possibility of a "mechanistic interpretation", as these will be the objects we are most directly interested in during this thesis, and we will study a more specific subset of such systems in Chapter 2, with the material in this chapter providing a more high level overview of dynamical systems and some approaches to modelling them. In the final chapter we return to a discussion of an additional class of such models, the class of "parameter driven" models, in Section 8.2.4 and discuss how they relate to the class of models we introduce in the next chapter.

1.1 Models of dynamical systems

The broadest definition of a dynamical system might be any system that changes over time. However, it is only when the patterns of change over time exhibit a certain kind of complexity that the technical apparatus of dynamic systems theory comes into its own, and so with greater specificity, we follow [Katok and Hasselblatt, 1995] and consider a dynamical system as being composed of the following key ingredients

Phase space A space whose elements represent possible states of the system.

Time A univariate variable which may be discrete or continuous, and which may extend either only into the future, or into the past as well as the future.

Time-evolution law In the most general setting this is a rule that allows us to determine the state of the system at each moment of time t from its states at all previous times.

It is typical to assume further that the time-evolution law depends only on the current state, and possibly the time variable, and this is the setting which we will also adopt. Systems for which the time-evolution law depends on the value of the temporal variable are referred to as non-autonomous.

The broad definition of a dynamic system carries through to the diverse range of possible candidate models for such a system; nevertheless, we can identify specific features that are likely to be realised by a successful description of a physical system by a dynamical model. Noteworthy features which will be reflected in the model developed in this thesis include

(i) A representation of the variations of a high dimensional system by trajectories constrained to a topologically rich, but lower dimensional, manifolds.

(ii) A distinction between structural parameters, and control parameters.

(iii) Decomposition of large complex systems into coupled subsystems with continuous circular causal influence among multiple subsystems.

While this is necessarily a reduced list, we can see the appearance of all of these features to varying degrees in many successful statistical models of dynamic systems. Indeed the importance of the first point extends far beyond the dynamical system setting and has been a cornerstone in the modern development of statistical inference for high dimensional datasets. In the context of statistical modelling, such representations are typically referred to as latent variable models, and the goal of these methods is to represent random variables in a, typically high dimensional, data space by the variations of a set of variables with lower intrinsic dimensionality, [Everitt, 1984]. The success of latent variable modelling in modern machine learning has inspired a great deal of research into extending the latent variable approach to the dynamic systems setting, including the modelling framework which we shall introduce and extend in this thesis.

The second item is more specific to dynamic systems models and expresses the idea that structurally similar systems may exhibit significantly different motions, and we view this heterogeneity as arising from different settings of the control parameters. Throughout this thesis, a motivating example will be human motion, and in this case, we might view the structural parameters as being joint lengths, muscle density and attachment points and various additional features of the human biology which for a given individual are taken to be fixed, and even across individuals have a modest variation. However on observing the swing of a golf club, a pirouette while dancing, or a stumble on an ice pavement by the same subject we will notice substantially different motions, under varying degrees of control, but all arising from the same physical structure. While for some scenarios the partitioning of the parameters into these two classes may seem artificial it seems clear that there are many cases for which this distinction is essential, and so must be given due regard during the construction of models for these systems. In general, the control function required to achieve a particular action will depend in a nontrivial way on the sequence of states itself, and therefore the entire system will display a complex feedback structure which would be daunting to attempt to model directly through autonomous state space equations. The introduction of control parameters, therefore, allows us to abstract away some of that complexity. We shall see how this is done in more detail when we introduce and extend, the latent force model in Chapter 2 which exhibits this partitioning of the model into a simple set of structural parameter governing the mechanistic model, and a flexible set of control parameters modelled by smooth Gaussian processes.

The final point is somewhat vague, but represents an essential conceptual step in the modelling process. Again the purpose is to allow us to avoid the necessity of an exhaustive description of the process at a microscopic level, and instead consider representative subsystems for which the interactions are better understood. Such constructions usually represent a tacit acknowledgement, even amongst those who favour a mechanistic approach to model building, that for complex systems a complete description is often out of reach. We view this final point as being distinct from the latent variable modelling approach, and arguably of having received less attention in modern statistical inference. Modern dimensionality reduction techniques will often seek to map a high dimensional state space to a lower dimensional space, which may be linear or nonlinear. However these methods do not typically address the possibility of any further decomposition of the latent space, and so the question of whether this space may itself be decomposed into simpler submanifolds or further global topological properties of the latent space manifold, are either not addressed or hard to discuss. For linear dimensionality reductions, the question is relatively uninteresting because the latent variable space is itself a Euclidean space, and the decomposition into the Cartesian product of Euclidean space adds little with regards to interpretability or interest to the model. However, in this thesis, we propose a method for embedding known geometric structure and so such a decomposition, and how information is shared between the subsystems, become a more interesting problem, and we describe this in more detail in Section 2.3.2. Unlike the latent variable model, this decomposition must be specified a priori, and so is most accurately viewed as a decomposition of the data space. A pressing direction of future research which we discuss towards the end of this thesis is the adaptation of the model introduced in this thesis to carry out the discovery of a dimension reducing latent variable space which itself admits a decomposition into a product of topologically distinct manifolds.

From these remarks, we may conclude that a successful dynamic explanation lies somewhere between a descriptive representation of events and a real causal explanation of why event unfold as they do; the first of these is the *data driven* paradigm, and the second is the mechanistic modelling approach. A useful model cannot merely be a description of the observed trajectory because a desirable feature of these models is that they have predictive power on observations away from the training data, that is they can generalise. On the other hand, even if a full causal description of a given dynamic system exists, it does not then follow that this description is immediately available and practical, and so some simplification must occur if these methods are to be of use to practitioners. This then is the conflict between the data-driven approaches and the mechanistic approaches motivating the hybrid modelling framework which we adopt and extend in this thesis. Before introducing the framework which we will make use of we first review some classes of dynamical models that have been of use in statistical inference, the review is necessarily brief and biased towards those models which will best inform the constructions later on in this thesis.

1.1.1 Ordinary differential equations

The models which we shall study in this thesis take the form of continuous time processes, throughout we shall denote the value at time t of a smooth continuous process taking values in $\mathbb{R}^K$ by $\mathbf{x}(t)$. We shall assume that the evolution of this smooth process is described by an initial value problem of the form

$$\frac{\mathrm{d}\mathbf{x}(t)}{\mathrm{d}t} = f(t, \mathbf{x}(t); \boldsymbol{\theta}) \qquad \mathbf{x}(t_0) = \mathbf{x}_0 \in \mathbb{R}^K, \tag{1.1}$$

where $f(\cdot)f$ is assumed to be some smooth function depending on the parameter $\boldsymbol{\theta}$, in those cases where the initial condition is also unknown, then this value can also be absorbed into the parameter vector. The parameter vector may be finite or infinite dimensional, and we will be particularly interested in the latter case in this thesis.

It is also regularly assumed that rather than directly observing the state $\mathbf{x}(t)$ we instead observe a collection of noisy observation $\{\mathbf{y}_i\}_{i=1}^{T}$ given by the additive noise model.

$$\mathbf{y}_i = \mathbf{x}(t_i) + \epsilon_i,$$

where the random variables ϵ_i are independent, identically distributed $\mathbb{R}^K$ valued random variables.

One of the most frequently encountered problems in this setting involves finding point estimates for the parameters describing the time-evolution equation in (1.1) from a set of observed trajectories. When it is possible to solve the IVP explicitly, then a natural choice of parameter estimation procedure is a minimisation of the least squares problem

$$\hat{\boldsymbol{\theta}} = \arg\min_{\theta} \sum_{i=0}^{T} \|\mathbf{y}(t_i) - \mathbf{x}(t_i; \boldsymbol{\theta})\|^2,$$

where $\mathbf{x}(t_i; \boldsymbol{\theta})$ denotes a solution of (1.1) with parameter $\boldsymbol{\theta}$. Since (1.1) only defines the solution to this equation implicitly it is usually necessary to use some numerical integration routine to estimate the unknown values $\mathbf{x}(t_i; \boldsymbol{\theta})$, and this must be done for each setting of the parameter during the optimisation procedure. This generally leads to nonlinear least squares problem and [Xue et al., 2010] have used the general theory of nonlinear least squares to show that under reasonable regularity conditions the nonlinear least squares estimator obtained using the numerical solutions of the ODE exhibits consistency and asymptotic normality.

Because of the high computational cost involved in estimating the parameters of these models using the numerical simulation strategy, there has been an interest in developing methods which avoid this step. Instead, starting from the early work of [Himmelblau et al., 1967, Varah, 1982], these methods use a two-step procedure starting by obtaining nonparametric estimators of the state and its trajectory. A typical choice is the non-parametric kernel estimator

$$\hat{\mathbf{x}}(t) = \sum_{i=1}^{T} (t_i - t_{i-1}) \frac{1}{b} K\left(\frac{t - t_i}{b}\right) \mathbf{y}_i,$$

here K iis a kernel function with bandwidth parameter b, see [Silverman, 1986]. These non-parametric estimators are then used as "plug-in" estimators to obtain the parameters by way of the explicit link between these variables as represented by (1.1) by minimising an objective function of the form

$$\hat{\boldsymbol{\theta}} = \arg\min_{\theta} \int_{t_0}^{t_T} \|\dot{\hat{\mathbf{x}}}(\tau) - f(\tau, \hat{\mathbf{x}}(\tau); \boldsymbol{\theta})\|^2 w(\tau)\mathrm{d}\tau,$$

for some choice of weight function $w(t)$. Recent examples and extensions of these methods include [Ramsay et al., 2007, Qi and Zhao, 2010, Gugushvili and Klaassen, 2012]. An important extension to the Bayesian setting by [Calderhead et al., 2009] which we shall consider in much more detail in Chapter 3 uses a Gaussian process to approximate the distribution of the unknown trajectory in the two-step procedure. The use of the two step process in this initial model leads to a methodological disconnect between the interpolator and the system dynamics, therefore [Dondelinger et al., 2013] considered an adaptation of this method which would allow updates of the interpolating estimator of the trajectory to be influenced by the current value of the parameters describing the dynamical systems structure of the model.

We also note that since any higher order ODE can be reduced to an equivalent system of first-order ODEs we shall, without loss of generality, focus only on first order ODEs. We discuss important issues regarding the extension of the methods in this thesis to partial differential equations (PDEs) in the final chapter.

1.1.2 Stochastic differential equations

In practice, the independent additive noise model typically assumed in the ODE model as discussed above is unrealistic. For complex systems the model may be misspecified, perhaps because a full mechanistic description is implausible, and an error at a given time is likely to feed back into the current value of the (misspecified) state, leading a cumulative error. It has therefore been of importance to consider more sophisticated error structures which can feedback directly into the evolution equation of the state variable itself.

One definition of a stochastic differential equation (SDE) is any system specified in terms of differences, finite or infinitesimal, with coefficients that are random variables. A more standard definition is a continuous time with the random fluctuations in the differences arising from a Brownian motion process. Acknowledging the sample path discontinuity of these equations they are typically presented in the form of an Itô SDE

$$\mathrm{d}\mathbf{x}(t) = f(t, \mathbf{x}(t))\mathrm{d}t + \sigma(t, \mathbf{x}(t))\mathrm{d}W_t,$$

where $\mathrm{d}W_t$ is a "white noise" process, which is viewed as a generalised derivative of a Brownian motion. Such systems may also be represented in integral form as

$$\mathbf{x}(t) = \int_0^t f(\tau, \mathbf{x}(\tau))\mathrm{d}\tau + \int_0^t \sigma(\tau, \mathbf{x}(\tau))\mathrm{d}W_\tau,$$

these methods have a rich mathematical literature, and many book length treatments are available including [Rasmussen and Williams, 2006, Øksendal, 2010]

Parameter estimation for SDEs is relatively straightforward from continuous time observations, or observations that are very densely sampled, in which case the combination of the independence of the increments and small-time Gaussian approximations to the transition distributions lead to easy to construct likelihood terms. See [Bishwal, 2008] for more details. For observations which are observed at a lower frequency, the situation is much more difficult, more recent approaches motivated by trying to complete the trajectory and then marginalising over the completion have been given in [Beskos et al., 2006, Kou et al., 2012]. In connection with this, it is interesting to note that the techniques we consider in Chapter 4 are also motivated by attempting to complete the trajectory and then appropriately integrating over the completion.

As we have indicated the most common use of SDEs in engineering and physics is as a method of incorporating random noise into classical models of dynamical systems in such a way that the noise can be allowed to affect the evolution of the state variable. There are undoubtedly some situations where the existence of a Brownian like component is an important part of the model, and there is an extensive literature on stochastic control problems, see for instance [Fleming and Rishel, 1975]. In this thesis, we shall typically consider smooth evolution equations, and for these smoothly guided systems, the diffusion structure is seen as being of reduced importance. Our chosen application to human motion again provides an example. It is reasonable that Brownian like vibrations would be a feature of such motion, and for these features to become increasingly pronounced for fatigued subjects, but at the same time modelling a golf swing, as we do in Chapter 7, at the level of a stochastic differential equation seems like an unnecessary introduction of mathematical complexity.

1.1.3 Linear dynamical systems

An important class of time series models occurs when the time-evolution equation is linear in the state variables. The general form of a continuous time linear dynamical system is given by

$$\frac{\mathrm{d}\mathbf{x}}{\mathrm{d}t} = \mathbf{A}(t)\mathbf{x} + \mathbf{S}(t)\mathbf{g}(t), \tag{1.2a}$$

$$\mathbf{y}(t) = \mathbf{H}(t)\mathbf{x}(t). \tag{1.2b}$$

The state of the system is represented by $\mathbf{x}(t)$, the introduction of the, potentially time-dependent, observation matrices $\mathbf{H}(t)$ allows for the variations of the higher dimensional output $\mathbf{y}$ to be explained by a lower dimensional variable, allowing one to adopt a latent variable modelling approach.

The case where $\mathbf{A}(t)$ is constant and $\mathbf{g}(t)$ is a white-noise process leads to the particular case of the Ornstein-Uhlenbeck SDE, which provides a classic example of an SDE which is explicitly solvable. Of greater relevance in this thesis will be the case where the functions $\mathbf{g}(t)$ are at least piecewise smooth. The most important instance of our work will be the case when this function is a smooth, vector-valued Gaussian process. This is precisely the latent force model of [Lawrence et al., 2006, Alvarez et al., 2009] which we provide an in-depth discussion of in Chapter 2.

For the class of models (1.4) we have the decomposition into a set of structural parameters given by $\mathbf{A}(t), \mathbf{S}(t)$ and $\mathbf{H}(t)$, as well as a set of control parameters given by the vector-valued function $\mathbf{g}(t)$. These systems have been well studied under the umbrella of control theory, and we provide some discussion of this literature in Section 2.4. However, one important difference which we discuss at this point, and related to our discussion of control parameters in the previous section is the underlying modelling perspective placed on these functions. In control theory, one is often concerned with designing a system to display some optimal property, in the statistical and machine learning setting however we are more interested in the inverse problem of learning the controls that could have realised an observed trajectory. Moreover, using these learned control parameters for further analysis such as classifying systems by their controls, or discussing qualitative properties by inspection of these functions.

We carry out an application of this program in Chapter 7 when we consider an application of dynamic systems modelling to motion capture data. It is often argued see for example [Bissacco, 2005], that the linear dynamic assumption is insufficient for such systems so that we cannot make immediate use of the model (1.4), and instead one must necessarily introduce more physically realistic nonlinear models.

While for this thesis we are primarily interested in statistical inference for continuous time models, the discrete time version of (1.4) has also been of much interest. In the discrete time setting this leads to a model of the form

$$\mathbf{x}_{n+1} = \mathbf{A}\mathbf{x}_n + \mathbf{w}_{n+1}, \tag{1.3}$$

where the errors $\mathbf{w}_n$ are given by a set of, independent, mean zero vector-valued Gaussian random variables. This class of models is often referred to as a linear Gaussian dynamical system [Bishop, 2006, Roweis and Ghahramani, 1999]. Such models have been well studied originating with the Weiner filtering problem and have lead to the development of the Kalman filter methods [Kalman, 1960a, Zarchan and Musoff, 2005]. While we do not use this model in this thesis, we do make use of the existence of efficient algorithms for inference in these models when we introduce an approximate inference method in Chapter 4. In particular, the Kalman filtering and smoothing methods allow for efficient calculation of the marginal and pairwise moments of a set of variables described by the system (1.5). In this case, the index set will correspond to an increasing approximation order, and not a temporal variable. The Kalman smooth equations are also sometimes referred to as the *Rauch-Tung-Striebel* (RTS) equations [Rauch et al., 1965].

1.1.4 Linear dynamical systems

An important class of time series models occurs when the time-evolution equation is linear in the state variables. The general form of a continuous time linear dynamical system is given by

$$\frac{\mathrm{d}\mathbf{x}}{\mathrm{d}t} = \mathbf{A}(t)\mathbf{x} + \mathbf{S}(t)\mathbf{g}(t), \tag{1.4a}$$

$$\mathbf{y}(t) = \mathbf{H}(t)\mathbf{x}(t). \tag{1.4b}$$

The state of the system is represented by $\mathbf{x}(t)$, the introduction of the, potentially time-dependent, observation matrices $\mathbf{H}(t)$ allows for the variations of the higher dimensional output $\mathbf{y}$ to be explained by a lower dimensional variable, allowing one to adopt a latent variable modelling approach.

The case where $\mathbf{A}(t)$ is constant, with negative real part of its eigenvalues, and $\mathbf{g}(t)$ is a white-noise process leads to the particular case of the Ornstein-Uhlenbeck SDE, which provides a classic example of an SDE which is explicitly solvable. Of greater relevance in this thesis will be the case where the functions $\mathbf{g}(t)$ are at least piecewise smooth. The most important instance of our work will be the case when this function is a smooth, vector-valued Gaussian process. This is precisely the latent force model of [Lawrence et al., 2006, Alvarez et al., 2009] which we provide an in-depth discussion of in Chapter 2.

For the class of models (1.4) we have the decomposition into a set of structural parameters given by $\mathbf{A}(t), \mathbf{S}(t)$ and $\mathbf{H}(t)$, as well as a set of control parameters given by the vector-valued function $\mathbf{g}(t)$. These systems have been well studied under the umbrella of control theory, and we provide some discussion of this literature in Section 2.4. However, one important difference which we discuss at this point, and related to our discussion of control parameters in the previous section is the underlying modelling perspective placed on these functions. In control theory, one is often concerned with designing a system to display some optimal property, in the statistical and machine learning setting however we are more interested in the inverse problem of learning the controls that could have realised an observed trajectory. Moreover, using these learned control parameters for further analysis such as classifying systems by their controls, or discussing qualitative properties by inspection of these functions.

We carry out an application of this program in Chapter 7 when we consider an application of dynamic systems modelling to motion capture data. It is often argued see for example [Bissacco, 2005], that the linear dynamic assumption is insufficient for such systems so that we cannot make immediate use of the model (1.4), and instead one must necessarily introduce more physically realistic nonlinear models.

While for this thesis we are primarily interested in statistical inference for continuous time models, the discrete time version of (1.4) has also been of much interest. In the discrete time setting this leads to a model of the form

$$\mathbf{x}_{n+1} = \mathbf{A}\mathbf{x}_n + \mathbf{w}_{n+1}, \tag{1.5}$$

where the errors $\mathbf{w}_n$ are given by a set of, independent, mean zero vector-valued Gaussian random variables. This class of models is often referred to as a linear Gaussian dynamical system [Bishop, 2006, Roweis and Ghahramani, 1999]. Such models have been well studied originating with the Weiner filtering problem and have lead to the development of the Kalman filter methods [Kalman, 1960a, Zarchan and Musoff, 2005]. While we do not use this model in this thesis, we do make use of the existence of efficient algorithms for inference in these models when we introduce an approximate inference method in Chapter 4. In particular, the Kalman filtering and smoothing methods allow for efficient calculation of the marginal and pairwise moments of a set of variables described by the system (1.5). In this case, the index set will correspond to an increasing approximation order, and not a temporal variable. The Kalman smooth equations are also sometimes referred to as the *Rauch-Tung-Striebel* (RTS) equations [Rauch et al., 1965].

1.1.5 Simulation based inference

The dynamical models we focus on this thesis typically take the form of a linear ODE, and as such they are relatively straightforward to solve numerically conditional on the parameters. In situations where this is possible then this suggests the possibility of a generative approach to model fitting by simulating various parameter settings, solving the model numerically, and then assessing the resulting model fit in the data-space. A class of such models is given by the so called method of approximate Bayesian computation (ABC). This approach to performing inference had been discussed at least as early as [Rubin, 1984], and applied to problems in population genetics in [Tavaré et al., 1997].

ABC methods work by simulating parameters $\boldsymbol{\theta}$ from the prior and then numerically solving them to produce a trajectory $\mathbf{x}(t) = \mathbf{x}(t;\boldsymbol{\theta})$, and if necessary this is further propagated to the data space to give $\mathbf{y}(t) = \mathbf{y}(t;\boldsymbol{\theta})$. If the simulated data is close in some suitable metric then we retain $\boldsymbol{\theta}$ as a sample from the posterior, otherwise we generate a new sample. This process is repeated until our desired sample size is achieved. There are technical details determining the choice of metric and determining the tolerance level for acceptance, although some results are available, [Beaumont et al., 2002, McKinley et al., 2009]. Furthermore it is often the case that rather than compare the trajectories directly in the data space one instead compares the distance of a set of summary statistics. Despite these complications the relatively simple conceptual nature of the model, and the familiarity many practitioners have with solving a complex model in their field given some set of parameters, these methods have become increasingly popular and we refer the reader to [Moral et al., 2012, Marin et al., 2012, T. McKinley, 2009] for more details.

In this work we shall be interested in models for which at least some of the variables take their values in a function space. Typically the set of functions which have a high probability under the posterior will occupy a very narrow region of the infinite dimensional parameter space, and therefore simulating such model from the prior is challenging, with a very low acceptance rate. Further mode such methods are only able to give an approximate sample from the posterior and so are not useful in deriving analytic approximations or motivating other deterministic methods. We discuss this issue in more detail in Chapter 3.

1.1.6 Non-parameteric estimation of dynamical systems

Here the goal is typically to estimate a, time-homogenous, discrete time-evolution map from a sequence of observations $\mathbf{x}_0, \mathbf{x}_1, \ldots, \mathbf{x}_T$ and where we write $\mathbf{x}_i$ as shorthand for $\mathbf{x}(t_i)$ for $i = 0, 1, \ldots, T$. Under the evolution law $\mathbf{x}_i = f(\mathbf{x}_{i-1})$, then successive pairs $(\mathbf{x}_{i-1}, \mathbf{x}_i)$ provide a natural choice of data for the construction of non-parametric estimators of the transition function. When the true map is continuous, and the sequence of iterates are dense in the state space then [Adams and Nobel, 2001] have shown that this map can be consistently estimated from the data by way of simple linear interpolation. These methods are appealing because they require very few assumptions, however as mentioned in [McGoff et al., 2015] the proofs make use of the ergodic theorem, and as such the speed of convergence may be hard to establish.

Of more direct relevance to our discussion is that such methods very much embody the data-driven paradigm, and there is no immediately obvious way to add prior structural knowledge into these estimation procedures. Therefore, in order for these methods to generalise there are likely to be situations which require a considerable quantity of data to perform adequately.

We may also be concerned by the fact that these methods do not have an immediate probabilistic interpretation, and so are hard to adapt to a Bayesian setting, incorporate missing data, or include as a component in more complex inferential frameworks. A method that allows the incorporation of these features is the Gaussian Process Dynamical Model (GPDM) [Wang et al., 2006]. For a state variable $\mathbf{y} \in \mathbb{R}^M$ and a K-dimensional state variable the time-evolution of the GPDM is described by the system

$$
\begin{aligned}
\mathbf{y} &= h(\mathbf{x}), \\
\mathbf{x}_n &= f(\mathbf{x}_{n-1}),
\end{aligned}
$$

for a smooth function $h : \mathbb{R}^K \rightarrow \mathbb{R}^M$ and smooth-time evolution map $f : \mathbb{R}^K \rightarrow \mathbb{R}^K$. Each of the component functions $h_m, m = 1, \ldots, M$ and $f_k, k = 1, \ldots, K$ are then given a GP prior.

Conditionally the distribution of Y has a straightforward Gaussian process prior. However, the distribution of the latent states is significantly more complicated. If $\mathbf{X} = [\mathbf{x}_0, \mathbf{x}_1, \ldots, \mathbf{x}_N]^{\top}$ then

$$p(\mathbf{X}) = p(\mathbf{x}_0) \frac{1}{\sqrt{(2\pi)^{NK} |\mathbf{K}_x|^K}} \exp\left(-\frac{1}{2} \operatorname{Tr}\left(\mathbf{K}_x^{-1} X_{out} X_{out}^{\top} \right), \right)$$

where $\mathbf{X}_{out} = [\mathbf{x}_1, \ldots, \mathbf{x}_N]^{\top}$, and the matrix K_x has entires given by $[\mathbf{K}_x]_{ij} = k(\mathbf{x}_i, \mathbf{x}_j)$, $i, j = 1, \ldots, N$ and $k(\cdot, \cdot)$ is some typically nonlinear kernel function. As a result the state variables appear nonlinear in the argument of the exponential so that the resulting distribution is intractable.

It is also not clear how to adapt this setup to irregularly spaced data, or how to add additional prior knowledge. Despite these caveats, [Wang et al., 2006, 2008] demonstrate the effectiveness of the GPDM at learning low dimensional representations with an interesting geometric structure. The model we introduce in this thesis may be viewed as a continuous time dynamical system constructed around a conditionally Gaussian random variable, and we compare our model with the GPDM in Section 8.2.3.

1.2 Scope of thesis

This thesis addresses the problem of developing flexible models for dynamical systems which can be utilised in those cases in which a full mechanistic description is hard to motivate, and the related problem of carrying out inference within these methods. Chapter 1 has provided a brief introduction to the existing literature on dynamic systems models, and how inference may be carried out within these methods. In Chapter 2 we focus our attention on a particular class of such models, these are the *latent force models* (LFM) introduced by [Alvarez et al., 2009]. The LFM represents a hybrid model combining a simple mechanistic representation with the flexibility offered by a Gaussian process forcing term. After introducing the LFM framework in this chapter, we propose our main extension, the latent force model with multiplicative interactions between the state variables and the driving GP terms, which we refer to as the *multiplicative latent force model.* We also provide some historical context for our extension including its appearance in the physics literature and approximations introduced there.

While the MLFM allows for the possibility of placing strong geometric constraints on the model trajectory, it is much harder to carry out inference than for the LFM setting. In this thesis we consider two methods of approaching this problem; the first uses a class of approximation methods for carrying out inference in ODE problems developed by [Varah, 1982, Ramsay et al., 2007] and adapted to a Bayesian setting in [Calderhead et al., 2009, Dondelinger et al., 2013]. While initially envisioned for carrying out inference on modest parameter vectors in nonlinear ODEs we demonstrate in Chapter 3 that these methods can be used in the infinite dimensional parameter setting, and that the use of linear ODEs simplifies the problem.

In Chapter 4 we consider an alternative approach that proceeds by motivating a series expansion of the trajectory and then approximating this expansion. The use of interpolants of the trajectory and its gradient necessarily involves the introduction of ambiguous tuning parameters, and it is hard to quantify the impact of these parameters on the resulting inference. On the other hand, the use of numerical integration as discussed above avoids the introduction of these spurious parameters, by attempting to construct a more faithful model we aim to combine the advantages of these approaches.

We treat the series expansion as the “complete data” for our model, and show that by conditioning on this augmented dataset we achieve tractable posterior distributions. However, this completion of the variable set leads to a vaster parameter space, and so in the next section, we consider methods that can be used to alleviate this problem.

In Chapter 5 we discuss variational approximations to both of the approximate densities introduced for our model. We first discuss how the expectation maximisation (EM) algorithm may be used to construct MAP estimates which better exploit the conditional independence structure of the methods we have introduced, this is particularly important for carrying out inference for the model introduced in Chapter 4 where the expectation step allows us to marginalise over the augmented variable set. We then provide details of the mean field variational Bayesian approximation to the densities we have introduced, once more the structure is naturally suggested by the conditional independence structure and leads to tractable distributions.

Our introduction of both methods is accompanied by a discussion about the scenarios in which we would expect each method to perform well. In Chapter 6 we provide further context to this by way of a comprehensive simulation study of the approximations we have introduced. While in general, we cannot recover the true conditional distribution of the MLFM, we can construct a good approximation to this distribution for the simple case of a dynamical system on the unit circle, and this allows us to carry out a comparison of the variational approximations with a ground truth distribution. In this chapter, we also consider an example with more complex geometry and discuss how we can assess the performance of our methods even in the absence of knowledge about the exact posterior distribution.

A heuristic motivation of the LFM framework is given by considering a marionette, in this instance, an articulated system with a fixed geometry can be manipulated by a small set of motions to realise a wide range of different trajectories. In Chapter 7 we demonstrate an explicit validation of this motivation by applying our MLFM to human motion capture data. We successfully demonstrate that the MLFM successfully models such a system, and by sharing information between different segments of the human body through a common set of latent forces we can achieve a superior predictive model compared to modelling each segment independently and that the learned forces allow for interesting qualitative analysis.

We conclude the thesis in Section 8.2 when we provide an overall review of the models we have introduced, as well as a discussion of important avenues for future research. Broadly speaking the future research directions can be divided into extended investigations of the asymptotic properties of both the MLFM and the approximations to it that we have considered, and also into increasing the range of applications of our approach; including a framework for combining significant dimensionality with automatic learning of a latent variable space with a rich topological structure.

Chapter 2

Latent force models

2.1 Introduction

Modern statistical inference for complex dynamical systems often seeks a balance between providing a comprehensive description of the data generating process, and an appeal to the *data-driven paradigm* whereby the model specification is kept suitably flexible so that, as far as is possible, statistical inference is driven by the observed data. The former of these two approaches are often referred to as the *mechanistic* approach and is common in the physical sciences including systems biology [Boogerd et al., 2013], chemistry [Olsson and Noé, 2017] and the geophysical sciences [Berliner, 2003]. In such cases, the claim is that it is possible to appeal directly to scientific knowledge to motivate a reasonable a priori class of possible data generating mechanisms. Typically these competing models will be indexed by values in a finite-dimensional parameter space. The data-driven approach, on the other hand, has been of increasing importance due to the growing abundance of large datasets where the volume of data, once combined with a suitably flexible model, is sufficient to allow essential patterns in the data to be revealed organically and not as an artefact of the chosen model. A common feature of this second class of models is the use of very large or infinite dimensional parameter spaces.

The recourse to data-driven methods is increasingly being preferred for those instances in which it would be infeasible to motivate a completely specified mechanistic model, and when a prediction is seen as being a more important goal than providing casual explanations. The choice to avoid the complicated modelling step is typically justified by appealing to the volume of data, and the claim that the quantity of data will be sufficient to discover important patterns and structure within the random process.

However, the concept of what constitutes a sufficiently large dataset is not well defined and is likely to itself be a function of the underlying system complexity, rather than a steadfast rule. As a consequence, purely data-driven approaches can often perform poorly when required to generalise and perform predictions away from the training data. This is in contrast with an appropriate mechanistic model which, if it has been well trained, will likely be able to demonstrate superior generalisation using the learned mechanistic interactions. Regarding these two approaches as opposite ends of a spectrum, it is natural to consider hybrid approaches that can reflect salient features of the dynamical system while at the same time being flexible enough to allow for a substantial contribution from the data.

An important factor underlying many successful modern statistical methods, and a desirable feature of the class of hybrid models in which we are interested, is the recognition that even for large and complex models much of the variation can often be well explained by a substantially smaller set of variables. This smaller dimensional representation is usually taken to be unobserved and related to the observed variables through some appropriate mapping from the lower dimensional space to the original data space, and typically this mapping must itself be discovered as part of the learning process. A notable case of this class of models is the semi-parametric latent factor model [Teh et al., 2005], which uses a set of linear transformations to mix a small set of latent GP variables to a higher dimensional output variable. Hybrid models of dynamic systems seek to adopt this latent variable modelling philosophy in such a way that the resulting model appropriately reflects the temporal nature of the data.

[Alvarez et al., 2009] have described one such class of models combining the latent variable modelling approach with relevant dynamic systems structure which they refer to as the linear *latent force model* (LFM). The history of this class of models may itself be viewed as having followed a trajectory from the mechanistic paradigm towards modern hybrid approaches. Initially, the model was introduced in [Lawrence et al., 2006] as a realistic mechanistic model of transcriptional regulation in gene networks, subsequent developments recognised the importance of the existence of a linear operator transforming the latent processes to the state space which we describe in Section 2.2.1. Recognition of this property allowed for the development of a class of GP models with a non-stationary kernel, thereby embedding some simple dynamical systems properties into the more general GP latent variable framework.

In this chapter, we first introduce the latent force framework by reviewing the specification of the LFM as a simple mechanistic model combined with additive Gaussian forces. A distinguishing feature of this model is the existence of a relatively simple expression for the pathwise solution which we present in Section 2.2.1. The existence of this explicit solution further leads to a joint Gaussian distribution for the trajectories in data-space and the latent forces, and so allows for the usual techniques of GP regression modelling [Rasmussen and Williams, 2006], to be used.

While the property of having Gaussian trajectories makes the inferential processes simpler, it is likely to be unsatisfactory for many datasets possessing known non-Euclidean geometric structure. Therefore, one of the main contributions of this thesis is to introduce an extension of the additive latent force model to allow for multiplicative interactions between the state and force variables, and we present this extension in Section 2.3. It is not hard to demonstrate that such a model will no longer have Gaussian trajectories, and we discuss how we may constrain the geometry of the support of the trajectories through a semi-parametric modelling approach that allows for rich topological structure to be embedded in the model while being driven by only a minimal set of latent forces.

Unfortunately, and in contrast to the LFM, this increased modelling complexity comes at the expense of tractable inference. One of the primary difficulties is that in general there is no closed form expression for the pathwise solution of the LFM with multiplicative interactions, and as a direct consequence no easy way to understand the transformation of the latent variable set to the data-space. We, therefore, discuss some existing methods that provide approximations to recovering the moments of these model in 2.3.1, although we shall find these only hold under very limiting assumptions, and require expensive methods to solve. It will be the problem of deriving a class of approximate methods of inference for the model introduced in this chapter which will be the principal focus of the following chapters.

2.2 Linear latent force models

The goal of hybrid modelling for dynamical systems may be described as the desire to construct a flexible class of models that transform a, typically small, set of driving latent forces, to the original data space through a linear mapping in a manner which also respects the temporal structure of the data. This aim is achieved in [Alvarez et al., 2009] by combining perhaps the most straightforward class of mechanistic models, linear ordinary differential ODEs with constant coefficient matrices, with an inhomogeneous forcing term which is taken to be modelled by a collection of GPs.

If $\mathbf{x}(t)$ is an $\mathbb{R}^K$-valued state variable with components $x_k(t)$, $k = 1, \ldots, K$ then we say that $\mathbf{x}(t)$ is described by a first order linear LFM if the dynamics of the sample paths are given by the system of ODEs

$$\frac{\mathrm{d}\, x_k(t)}{\mathrm{d}\, t} + D_k x_k(t) = B_k + \sum_{r=1}^{R} S_{rk} g_r(t), \qquad k = 1, \ldots K, \tag{2.1}$$

where D_k, B_k are real-valued scalars for $k = 1, \ldots, K$ and $\{g_r(t)\}_{r=1}^{R}$ is a set of R real-valued smooth Gaussian processes with mean zero and a kernel functions $k_r(t, t'), r = 1, \ldots, R$ defined over $\mathbb{R} \times \mathbb{R}$ such that for arbitrary finite samples the corresponding Gram matrix is positive definite. The kernel functions of the GP variables are chosen so that the resulting sample paths are almost surely continuous, and this allows us to interpret (2.1) as an ODE rather than requiring the mathematical development of SDEs, necessary properties of the kernel function to satisfy this assumption are discussed in [Adler and Taylor, 2007].

Equivalently we may consider the vector-valued process $\mathbf{x}(t) \in \mathbb{R}^K$ which satisfies the ODE in matrix-vector form given by

$$\frac{\mathrm{d}\, \mathbf{x}(t)}{\mathrm{d}\, t} = -\mathbf{D}\mathbf{x} + \mathbf{b} + \mathbf{S}\mathbf{g}(t), \tag{2.2}$$

where $\mathbf{D}$ is a $K \times K$ diagonal matrix with entries $\mathbf{D}_{kk} = D_k$, the K-vector $\mathbf{b}$ is formed by the elements $B_k, k = 1, \ldots, K$, $\mathbf{S}$ is a $K \times R$ matrix and $\mathbf{g}(t)$ is an R-dimensional real valued stochastic process with elements given by the independent latent forces $\{g_r(t)\}_{r=1}^{R}$.

The rectangular matrix $\mathbf{S}$ acts to distribute linear combinations of the latent forces into each component of the dynamics of the ODE and will be referred to as the "sensitivity matrix". The sensitivity matrix acts to control the dependency structure between different output dimensions and it should be stressed that under the diagonal matrix specification in (2.1) it is the only way in which the dynamics of one particular component $x_k(t)$ may be linked to the dynamics of another, distinct, component. It is for this reason that while such a system can display some interesting dynamics, it is still a very simplistic model of a physical system.

We shall discuss in Section 2.4 the concept of controllability and reachability for a controlled dynamic system. Succinctly this concept discusses whether for a given system structure there is some set of latent forces that would allow any collection of points to be interpolated by a trajectory from the system. Reachability implies there is some maximal model that will interpolate the observed data, but that in doing so a great deal of the explanatory burden will be placed on the latent forces. As such under this diagonal specification, we are likely to experience overfitting. In particular simple first-order interactions between variables which could be described by a single parameter must now be modelled by an infinite dimensional parameter, and the resulting model is likely to generalise poorly. Even if overfitting does not occur there remains the issue of model parsimony where we now have simple interactions being modelled by an infinite dimensional parameter. As such we would like to extend this model in such a way as to allow for more richer specifications. This point provides further motivation for the class of models we shall introduce in the following section.

An alternative method of preventing the problem we have just described would be to allow for more general forms of the coefficient matrix $\mathbf{A}$. While the assumption of diagonal coefficient matrices may seem initially constricting it is worth noting that if we consider the more general model

$$\frac{\mathrm{d}\mathbf{y}(t)}{\mathrm{d}t} = \mathbf{U}\mathbf{y}(t) + \mathbf{S}\mathbf{g}(t), \tag{2.3}$$

where $\mathbf{U}$ is no longer constrained to be diagonal, but is instead assumed to be similar to a diagonal matrix. From this assumption we have the decomposition $\mathbf{U} = \mathbf{P}\mathbf{D}\mathbf{P}^{-1}$ for some invertible square matrix $\mathbf{P}$ and diagonal matrix $\mathbf{D}$, which may be complex valued. If we consider the transformed variable $\mathbf{x} = \mathbf{P}^{-1}\mathbf{y}$ then the transformed system will have an evolution equation given by

$$\frac{\mathrm{d}\mathbf{x}(t)}{\mathrm{d}t} = \mathbf{D}\mathbf{x} + \mathbf{P}^{-1}\mathbf{S}\mathbf{g}(t), \tag{2.4}$$

which gives one way of constructing such models and of moving between models with richer, but still linear, state interactions. In general, such decomposition may not exist, or the resulting matrices are typically complex valued. The complex case is not overly problematic, but it would require doubling the state space by identifying $\mathbb{C}^K$ with $\mathbb{R}^{2K}$. More problematic is that in general this diagonalising transformation is unknown and must also be discovered during the learning process.

These comments suggest that rather than constructing the LFM in the ambient dataspace it may be worthwhile to include an additional layer to this model and assume that the possibility higher dimensional data is modelled by the more general linear LFM (2.3). When it comes to representing a solution of the LFM, it is straightforward to present the formal solution of (2.3), and the same arguments will show that this is a GP, but in the non-diagonal case it becomes much harder to evaluate the kernel function analytically. Even if this problem can be overcome, and a diagonalising transformation of the latent force learnt simultaneously, the resulting process will still be a Gaussian process. In some applications, this assumption is likely to be implausible, and this will undoubtedly be the case where the process is not supported on a linear vector space, and we discuss this extension in Section 2.3.

Before moving on to discuss the solutions of the LFM we provide a few more remarks on the interpretation of the latent force framework. In [Alvarez et al., 2009] the comparison is made to a marionette where an extensive range of complex motions are generated by the smooth manipulation of a small number of strings. The resulting motion is then given by the guided, exogenous manipulation of the handles, and the fixed attachment point of the strings, which may be viewed as an endogenous, structural component of the system. This is an important motivation for this class of models, and we provide a literal interpretation of this imagery when we use our extension of the LFM to fit motion capture data in Chapter 7. It also suggests an essential feature of the driving forces, in that they represent guided, intentional motions, and should not be viewed as nuisance parameters to be marginalised, and certainly not as the memory-less Markov process interpretation which is common in the SDE setting. This is not to argue that marginalisation does not have a computationally useful role, and indeed we shall see that the flexibility to marginalise out the forces in the LFM allows the additional parameters to be learned without jointly optimising the latent force variables. This is a very different interpretation to that typically present in the SDE setting, a classically important example of a SDE is the version of (2.1) where the latent force variables are replaced with delta correlated white noise forces. In this view the white noise forces are typically regarded as nuisance components, their existence is of physical relevance, but the individual sample paths of these trajectories are not of any interest in themselves, and so the focus is on marginalising over the contribution of these latent forces.

2.2.1 Solution of the LFM

In general, the trajectory of dynamic systems with random components will be defined only implicitly be their evolution equations, and therefore in order to carry out inference, it is usually necessary to provide an explicit representation of the trajectory as a transformation of the fundamental random processes. Ideally, this is done by being able to write an explicit closed form solution. The importance of a closed form expression for the state variables as a transformation of the underlying latent variables to the subsequent construction of the conditional distributions will be a recurring theme in this thesis. In Chapter 3 we shall describe a method that circumvents this step, and in Chapter 4 we shall describe a method constructed around a series expansion of such a transformation.

One of the attractive features of linear dynamic systems of the form (2.2) is the existence of an explicit solution

$$\mathbf{x}(t) = e^{-\mathbf{D}t}\mathbf{x}_0 + \int_0^t e^{-\mathbf{D}(t-\tau)} \left(\mathbf{b} + \mathbf{S}\mathbf{g}(\tau)\right) \mathrm{d}\tau, \tag{2.5}$$

which may be checked by directly differentiating this expression, or see [Arnold, 1973]. For arbitrary matrix $\mathbf{D}$, not necessarily diagonal, the matrix exponential in (2.5) is defined by the absolutely convergent power series $e^{\mathbf{A}} := \sum_{k=0}^{\infty} \mathbf{A}^k/k!$. As we briefly remarked above the general solution (2.5) would allow us to relax the diagonal constraint of the model (2.1), and allow for richer interactions between the state variables. However, the matrix exponential in the integral of (2.5) makes the resulting expression significantly harder to evaluate when deriving expressions for the mean and covariance functions, and it is for this reason that the diagonal constraint is imposed.

Restricting ourselves to the case where $\mathbf{D}_k$ is a diagonal matrix (2.5) simplifies to the single components

$$x_k(t) = e^{-D_k t}x_k(t_0) + \frac{b_k}{D_k}(1 - e^{-D_k t}) + \sum_{r=1}^{R} \mathcal{L}_{kr}[g_r](t), \tag{2.6}$$

for $k = 1, \dots, K$ and where we are following the notation established in [Alvarez et al., 2009] by drawing particular attention to the linear convolution operator $\mathcal{L}_{kr}$ defined by

$$\mathcal{L}_{kr}[g_r](t) \triangleq S_{kr} \exp\left(-D_k t\right) \int_0^t \exp\left(D_k \tau\right) g_r(\tau)\mathrm{d}\tau. \tag{2.7}$$

The operators $\mathcal{L}_{rk}$ are convolution type operators representing the contribution to the trajectory of the kth state variable arising from the rth latent force. This is an integral of an exponentially weighted Gaussian process, and so from the closure of Gaussian random variables under linear operators, it follows that $\mathcal{L}_{rk}[g_r](t)$ will itself be a Gaussian process. By taking the expectation through the integral, we have that the mean of this process is given by

$$\mathbb{E}\left[x_k(t)\right] = e^{-D_k}\mathbb{E}\left[x_k(t_0)\right] + \frac{b_k}{D_k}(1 - e^{-D_k t}). \tag{2.8}$$

Typically we assume that $\mathbb{E}\left[x_k(t_0)\right] = 0$ and that the initial condition is independent of the model parameters when deriving expressions for the covariance. To calculate the covariance, we shall make use of the tensor product notation from [Lawrence et al., 2006] and so write the covariance as

$$\begin{aligned}
\operatorname{Cov}\left\{\mathcal{L}_{kr}[g_r](t), \mathcal{L}_{k'r'}[g'_r](t')\right\} &= \mathcal{L}_{kr}[g_r] \otimes \mathcal{L}_{k'r'}[g'_r](t,t') \\
&\triangleq S_{kr}S_{k'r'}e^{-D_k t - D_{k'}t'} \int_0^t \int_0^{t'} e^{D_k\tau + D_{k'}\tau'} \operatorname{Cov}\left\{g_r(t)g_{r'}(t')\right\} \mathrm{d}\tau\mathrm{d}\tau' \\
&= \delta_{rr'}S_{kr}S_{k'r'}e^{-D_k t - D_{k'}t'} \int_0^t \int_0^{t'} e^{D_k\tau + D_{k'}\tau'} k_r(\tau,\tau')\mathrm{d}\tau\mathrm{d}\tau',
\end{aligned} \tag{2.9}$$

where $\delta_{rr'}$ is the Kronecker delta with $\delta_{rr'} = 1$ if $r = r'$ and 0 otherwise, and appears because of the prior assumption of independence between the latent forces. For some popular choices of kernel function the double integral (2.9) can be evaluated explicitly, or at least approximated by quickly converging series approximations. An important example is the case where the kernel is chosen to be the popular radial basis function kernel, and a closed form expression for this case is given in [Alvarez et al., 2009] in terms of the 'erf' function related to the cumulative distribution function of a standard Gaussian random variable, implementations of which are typically available in common numerical libraries.

Assuming it is possible to evaluate the integral in (2.9) analytically then we are able possible to realise the marginal distribution of the trajectories by integrating out the contribution from the latent forces. After marginalising over the latent forces we can construct the likelihood term $p(\mathbf{x} \mid \boldsymbol{\theta})$ where the parameters $\boldsymbol{\theta}$ include the model structure parameters $\{\mathbf{D}, \mathbf{S}, \mathbf{b}\}$ as well as kernel hyperparameters. This allows us to estimate these values through optimisation of this likelihood term without having to jointly optimise for the latent forces which will lead to a computationally more efficient method. Once an appropriate optimisation routine has been performed we can then return the conditional posterior of the latent forces for given structural parameters, this allows us to efficiently learn the model without reference to the infinite dimensional latent force parameter, but then return the conditional distribution of the latent force terms if desired.

The net result is that the specific solution (2.5) leads to tractable Gaussian processes regression models with a mean and covariance parametrised by the kernel hyperparameters, as well as a set of structural parameters arising from the dynamic system, $\mathbf{S}, \mathbf{D}, \mathbf{b}$. From inspection of (2.4) is clear that the only interactions between the state variables are through the common latent force variables, and the sensitivity matrix $\mathbf{S}$ controls the topology of these interactions. The entries D_k, $k = 1, \ldots, K$ of the diagonal matrix K serve to determine the stability of the system, and are analogous to the role played by the eigenvalues in the usual interpretation of systems of ordinary linear differential equations although in this case we are limited to the case of strictly real eigenvalues.

While this is undoubtedly a critical extension from the usual class of Gaussian process regression models the range of dynamic systems behaviour successfully encoded is still limited, in effect we have a nonstationary covariance function which may, or may not, have a stationary limit depending on the eigenvalues of the matrix $\mathbf{D}$. More generally we note that the spectrum of the matrix $\mathbf{D}$ determines the transients in the mean function (2.8), and the exponential decay or growth of the variance in (2.9), we may conclude that the matrix $\mathbf{D}$ determines overall stability properties of the system, while the sensitivity matrix acts to modulate the interactions between individual components.

It should also be noted that it is possible to extend this model to higher order systems of differential equations. For example we could do so quite naturally by identifying any nth order differential equation in a single variable with an equivalent system of n first order differential equations, unfortunately, to preserve analytic tractability we need to maintain the diagonal structure of (2.2) which may limit the class of higher order systems we can consider. Nevertheless [Alvarez et al., 2009] do demonstrate the extension to the second order case with radial basis function kernels placed on the latent forces, and demonstrates that this model can exhibit a much richer class of dynamic motions than the first order model. Ultimately any higher order model will necessarily still be Euclidean supported, and therefore in order to get a richer set of dynamic systems behaviour, and in particular a richer set of geometric constraints, it would be necessary to consider nonlinear differential equation models or the extension we introduce in the following section.

2.3 Multiplicative latent force models

While being able to view the linear latent force model of the previous section as a particular instance of the standard GP regression model is appealing from an inferential point of view; the additive inhomogeneity of the LFM leads to a Gaussian process which is necessarily supported on a vector space. For many dynamic systems with applications in engineering and physics, this assumption may be inappropriate. Such systems of practical importance include time series of circular or directional data [Mardia and Jupp, 2000], tensor-valued data [Kagan, 1992, Xie et al., 2010] and various other datasets which possess a high degree of geometric structure, and where this geometric structure is known a priori. For all of these cases if we have a suitably dense sample of observed data, then the Gaussian trajectory assumption may be acceptable by considering local linear approximations to more complex geometries, however when data is sparse relative to the model structure, and in particular its geometry, this assumption becomes inadequate.

Therefore in many of these cases, it would be desirable to embed known geometric structure within the modelling framework, and in this section, we propose an extension of the latent force framework to allow for models with potentially strong geometric constraints. Our construction will retain the essential ingredients of the LFM framework; specifically, we are going to retain the property of being a linear ODE with the time-dependent behaviour arising from the fluctuations of a set of R independent smooth Gaussian processes. Where our proposed model will differ is in allowing for multiplicative interactions between the latent state variables, and the latent force variables, which we do by specifying an evolution equation of the form

$$\frac{\mathrm{d}\mathbf{x}(t)}{\mathrm{d}t} = \mathbf{A}(t)\mathbf{x}(t), \qquad \mathbf{A}(t) = \mathbf{A}_0 + \sum_{r=1}^{R} g_r(t) \cdot \mathbf{A}_r, \tag{2.10}$$

whereas in the previous section $\mathbf{x}(t)$ is a vector-valued process in $\mathbb{R}^K$. The geometric structure of this model emerges through the choice of the set of coefficient matrices $\{\mathbf{A}_r\}_{r=0}^R$, where each $\mathbf{A}_r$ is a $K \times K$ matrix with real valued entries. The time dependent coefficient matrix $\mathbf{A}(t)$ is formed of random linear combinations of scalar GPs multiplied by the set of coefficient matrices and so will itself be a matrix-valued, GP. We denote the support of vectorisation of the matrix-valued GP by $\mathcal{A} \subset \mathbb{R}^{K^2}$, and it is determined by the affine space

$$\mathcal{A} = \left\{ \mathbf{v} \in \mathbb{R}^{K^2} \;:\; \exists\, \mathbf{a} \in \mathbb{R}^R \text{ s.t. } \mathbf{v} = \operatorname{vec}(\mathbf{A}_0) + \sum_{r=1}^{R} a_r \operatorname{vec}(\mathbf{A}_r) \right\}.$$

There is no reason in general for this to coincide with the full K^2 dimensional space, and indeed we shall see that in the case we are interested in this space is typically of a much smaller dimension because of the structure of the set of coefficient matrices.

The choice of the structure matrices $\{\mathbf{A}_r\}_{r=0}^R$ will allow for the possibility of embedding strong geometric constraints on the space of possible trajectories. This is most easily seen by introducing the concept of a Lie group, and its associated Lie algebra, see [Hall, 2015, Helgason, 1962, Kobayashi and Nomizu, 1963]. We will make little use of the rich theory of such manifolds but what is important for our purposes is the general result, [Iserles and Nørsett, 1999], that if G is some matrix Lie group with Lie algebra $\mathfrak{g}$ and the coefficient matrix is constructed so that $A(t) \in \mathfrak{g}$ for all t in some interval $[0, T]$ then the initial value problem

$$\frac{\mathrm{d}\mathbf{x}(t)}{\mathrm{d}t} = \mathbf{A}(t)\mathbf{x}(t), \qquad \mathbf{x}_0 \in G, \tag{2.11}$$

is constrained to lie in the group for all $t \in [0, T]$. We may also consider the case where the data space is a vector space on which the group G acts by solving the model with initial condition $\mathbf{x}_0 = I$, where I is the identity element of the group, and in the case of matrix Lie groups corresponds to the usual identity matrix. Solving (2.11) with initial condition I gives a trajectory on the group which leads to a time-varying action so that at each time point the trajectory in the data space is given by the action of a particular member of the group G on the initial condition. This allows us to consider not only dynamic systems on groups but also vector-valued dynamic systems realised as a random action of this group on the vector space.

Example: MLFM for rotation valued data

A notable example of this setup is the case when the group, G, is the group of rotations of a D-dimensional Euclidean space, typically denotes by $SO(D)$, in which case the resulting trajectories will be given by continuous rotations of the initial condition leading to smooth paths with fixed radius from the origin. If we consider the simplest case of a point in $\mathbb{R}^2$ undergoing a smooth, stochastic rotation then we can model this with a MLFM of the form

$$\frac{\mathrm{d}}{\mathrm{d}t} \begin{bmatrix} x(t) \\ y(t) \end{bmatrix} = g(t) \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} x(t) \\ y(t) \end{bmatrix}. \tag{2.12}$$

The coefficient matrix in (2.12) is the canonical representation of the Lie algebra $\mathfrak{so}(2)$ of "infinitesimal rotations" of a point $\mathbb{R}^2$. At each instance the latent force then modulates this rotation – the magnitude of the latent force defining how fast the rotation occurs, and the sign representing whether this rotation is in a clockwise or anti-clockwise direction.

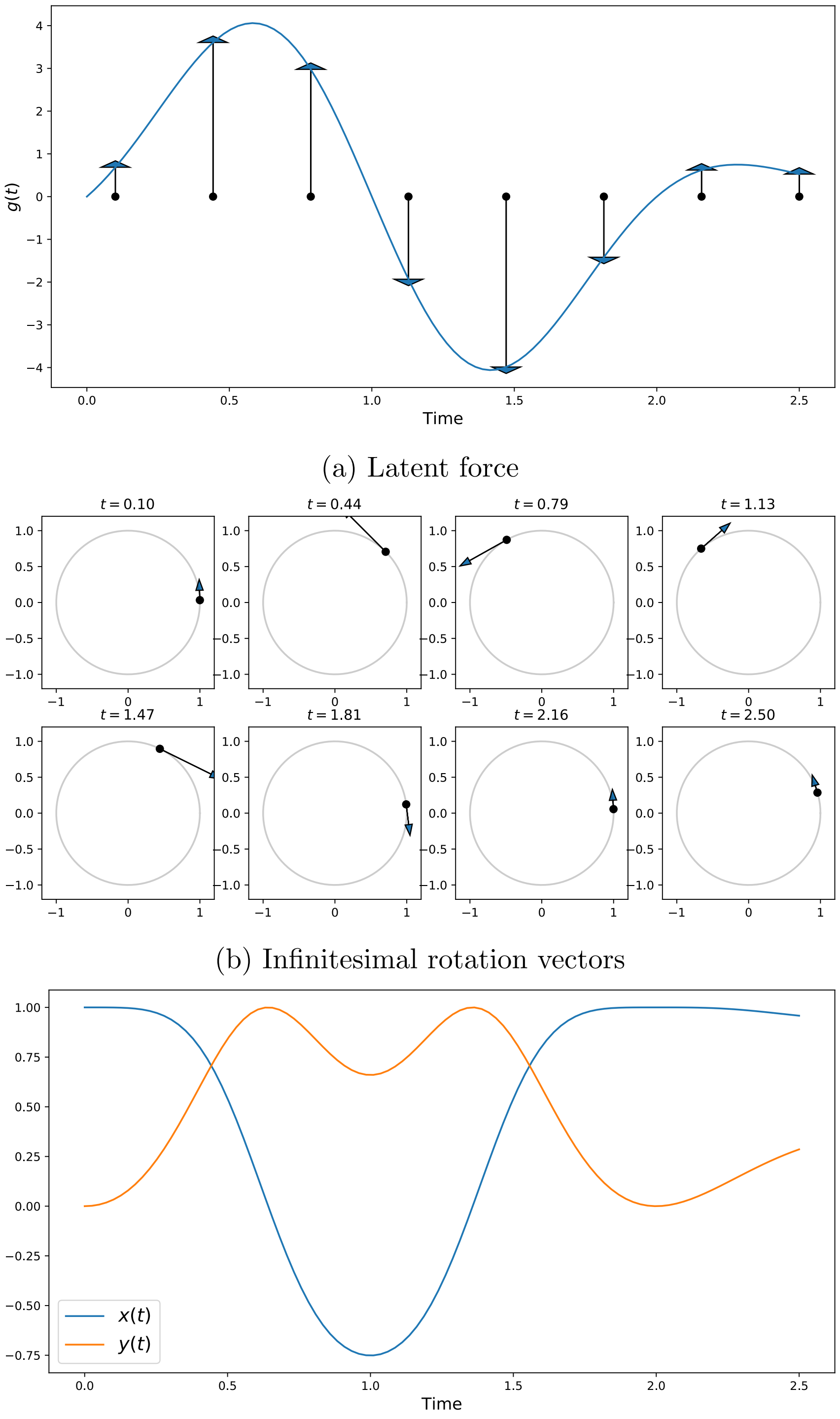
We represent the MLFM dynamics graphically in Figure 2.1; first the latent force used to drive the system is displayed in Figure 2.1a and for discrete set of points we also draw a collcetion of vectors of length denoted by the value of the forcing function at that time. These vectors are identified with elements of the abstract Lie algebra $\mathfrak{so}(2)$ to produce a set if "infinitesimal actions" determined by the right-hand side of (2.12), and we represent these graphically as the direction vectors in the phase space portraits given in Figure 2.1b which present the direction and magnitude of an infinitesimal rotation of the current state of the system. The complete trajectory in Figure 2.1c is then given by the composition of all of these infinitesimal actions, and so we can observe that starting from the initial condition $(1,0)^\top$ the initially positive latent force leads acts as an accelerating rotation in an anti-clockwise direction, before the latent force returns to zero when the motion reaches a stationary point, after which the point moves back in a clockwise direction. We consider the rotation example in further detail as part of Chapter 6, and we continue to the case of fitting paths on the group itself in Chapter 7.

Dimensionality reduction

In keeping with the underlying philosophy of the latent variable framework we would like the number of latent force variables R to be modest, however in (2.10) the dimension of R is related directly to the size of the set $\{A_r\}_{r=0}^{R}$, and so by extension the dimension of the Lie algebra generated by this set. It follows that the number of latent forces in the specification (2.10) is tied to the geometry of the data space, and therefore if we wish to increase the dimension of the Lie algebra we require a corresponding increase in the number of latent force variables. As a latent variable model, this property is unsatisfactory, and we would like to ensure a rich space of attainable trajectories with only a modest number of latent force variables. To achieve this we further extend the model by allowing the structure matrices themselves to be determined hierarchically by linear combinations of a fixed set of basis matrices $\{\mathbf{L}_d\}_{d=1}^{D}$, which we assume to be a basis for a Lie algebra, the structure matrices are now given by linear combinations of these basis elements

$$\mathbf{A}_r = \sum_{d=1}^{D} \beta_{rd}\mathbf{L}_d, \tag{2.13}$$

where we have introduced the $(R+1)\cdot D$ parameters β_{rd} which we store in the array $\mathbf{B}$ with $\mathbf{B}_{rd} = \beta_{rd}$. This allows us to consider a D dimensional Lie algebra, and its associated Lie group, independently of the number of forces. So allowing a much broader range of possible trajectories driven by a small number of forces, this is however at the expense of identifiability. In this initial specification (2.10) each force may be interpreted as modulating the infinitesimal action of the associated structure matrix $\mathbf{A}_r$. If the matrices $\mathbf{A}_r$ are chosen so that they are independent in the linear algebraic sense, then we can interpret each term $\mathbf{A}_r \cdot g_r(t)$ as infinitesimal independent actions of a one-parameter subgroup corresponding to this basis Lie algebra element on the data. The richer specification loses this interpretability where with $R < D$ it will no longer be the case that the set $\{A_r\}_{r=1}^{R}$ is linearly independent. Later on this section, we present a more extended discussion on the topic of identifiability, however from a machine



(a) Latent force

(b) Infinitesimal rotation vectors

(c) MLFM trajectory

Figure 2.1: Representation of the MLFM with coefficient matrix lying in the Lie algebra of the rotation group $SO(2)$. (a) The latent force determining the direction and magnitude of the infinitesimal rotations. (b) Phase space view of the infinitesimal rotation determining the dynamics of the MLFM, the trajectories are constrained to the circle S^1 with radius determined by the initial point $(1, 0)^\top$. (c) Resulting trajectory as a composition of all of the infinitesimal rotations.

learning perspective with the greater emphasis placed on prediction the importance of identifiability is diminished in comparison with the mechanistic approach. In practice for complex systems, the added flexibility of (2.13) seems preferable to the increase in the number of forces that may be required in (2.10) to achieve comparable predictive performance.

Comparison with the LFM On the same theme of dimension reduction it is important to note that one could, in principle, interpolate any given set of points generated by dynamical system on R^D, and in particular any system constrained on a submanifold of $\mathbb{R}^D$, with a sufficiently complex latent force model. Given these comments it is worth considering what we gain by instead allowing for multiplicative interactions if a sufficiently complex LFM may be able to fit a particular realisation of this system. However, given the restriction to simple diagonal systems in Section 2.2, all of the work for this interpolating system must be done by the latent force functions – therefore multiplicative interactions allows for more of the work do be done by the structure of the flow function of the dynamical system, with the latent forces in the MLFM allowing for perturbations of the dynamics within a constrained geometry, in the LFM the latent force would also have to do all of the work to constrain the geometry, as well as accounting for interesting deviations within this geometry. In Figure 2.2 we display the trajectories of two systems on R^3 forced by the same latent force, we observe the distinct qualitative differences in the trajectories between the two systems. The multiplicative extension of the MLFM is able to encode the geometric constraints into the system dynamics, with the forcing function adding additional perturbations on top of this structure. Indeed from the discussion preceeding (2.11) we observe that even in the absence of forcing the trajectories would be constrained to the sphere S^2 as in Figure 2.2c, whereas for the first order LFM the only encoded dynamical system action is the exponential damping of a given input force as observed in Figure 2.2d.

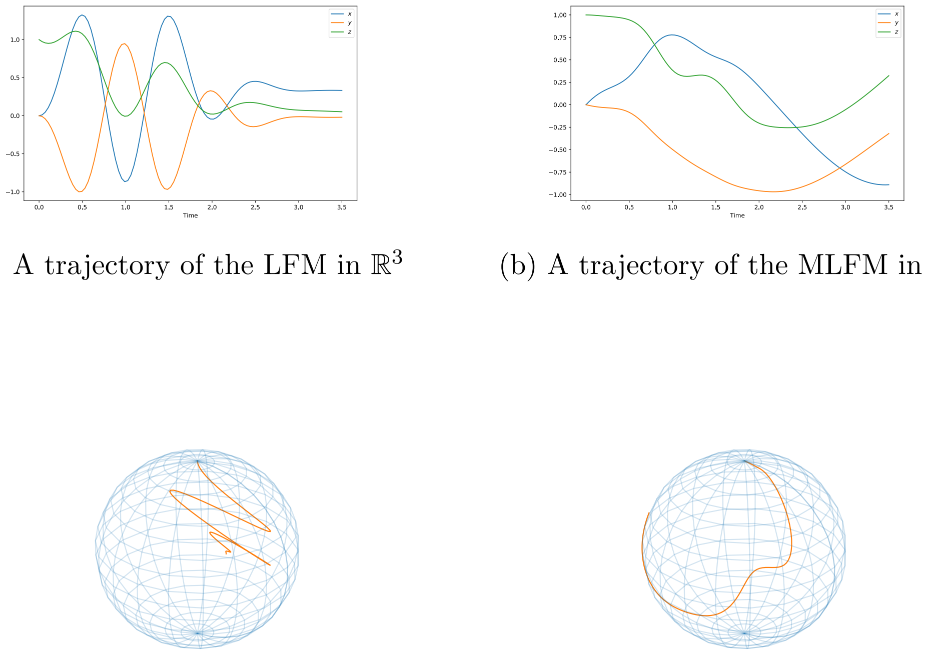
All of the discussion above should make it immediately clear that we can no longer expect the solutions to our multiplicative latent force model (2.10) to possess the tractable GP regression model structure. In the next section, we present a series expansion of the pathwise solution to the MLFM, as well as an approximation to the moments analogous to those presented in Section 2.2.1 which hold under a restrictive set of assumptions.

2.3.1 Solution of the MLFM

We make the same assumptions as for the LFM, that is the kernels for the GP terms are chosen so that elements of the reproducing kernel Hilbert space (RKHS) associated with each of these kernels is almost surely smooth. If we consider the stochastic components $\{\mathbf{x}_0, \{g_r(t)\}_{r=1}^{R}, \mathbf{B}\}$ as being elements of a probability space $(\Omega, \mathcal{F}, \mathbb{P})$ then we assume that for $\omega \in \Omega$ we have, almost surely, that the matrix-valued process $\mathbf{A}(t) = \mathbf{A}(t;\omega)$ is smooth. From this, it follows that for a given ω we can conclude that the differential equation

$$\frac{\mathrm{d}\mathbf{x}(t)}{\mathrm{d}t} = \mathbf{A}(t;\omega)\mathbf{x}(t;\omega), \tag{2.14}$$

has a unique solution on a compact interval $[0,T]$ by standard existence and uniqueness theorems for ODEs [Arnold, 1973]. It is reasonable to then ask whether we can conclude that a solution exists after marginalising over the latent GP variables, the connection coefficients or both. This was possible for the latent force variables in the LFM because the formal solution (2.5) contained a simple linear transformation of the latent GP



(a) A trajectory of the LFM in $\mathbb{R}^3$

(b) A trajectory of the MLFM in $\mathbb{R}^3$

(c) Phase portrait of the LFM in $\mathbb{R}^3$

(d) Phase portrait of the MLFM in $\mathbb{R}^3$

Figure 2.2: (a) A typical example of a single trajectory of the LFM with $D = 3$. (b) A typical example of a single trajectory of the MLFM with Lie algebra $\mathfrak{so}(3)$ with initial condition $(0, 0, 1)^\top$ and with the same forcing function for both systems. (c) Phase portrait of the trajectory represented in (a), and (d), phase portrait of the trajectory displayed in (b). The trajectories of a stable LFM quickly converge to zero unless forward away and in this instance are contained within the unit sphere (c), while those of the MLFM (d) are constrained to lie on the surfaces of the sphere S^2, and would be so even in the absence of forcing.

terms, and so was easily seen to be Gaussian.

We shall see in Chapter 4 that a formal solution to (2.14) may be given by the series expansion

$$\begin{aligned}\mathbf{x}(t) = \Bigg(\mathbf{I} + &\int_{t_0}^{t} \mathbf{A}(\tau)\mathrm{d}\tau \\ &+ \int_{t_0}^{t} \mathbf{A}(\tau_1) \int_{t_0}^{\tau_1} \mathbf{A}(\tau_2)\mathrm{d}\tau_2\mathrm{d}\tau_1 \\ &+ \int_{t_0}^{t} \mathbf{A}(\tau_1) \int_{t_0}^{\tau_1} \mathbf{A}(\tau_2) \int_{0}^{\tau_2} \mathbf{A}(\tau_3)\mathrm{d}\tau_3\mathrm{d}\tau_2\mathrm{d}\tau_1 + \cdots \Bigg)\mathbf{x}_0. \qquad (2.15)\end{aligned}$$

Typically we shall assume independence of the initial condition and the process $\mathbf{A}(t)$, so that it is the presence of the nested integrals of products of random variables that is the most problematic when attempting to marginalise over the contributions of the latent forces. The expansion (2.15) is a complex nonlinear transformation of the latent variables, and even in the simplest case when the matrix-valued functions $\mathbf{A}(t)$ commute leading to the simplification

$$\begin{aligned}\mathbf{x}(t) &= \Bigg(\mathbf{I} + \int_{t_0}^{t} \mathbf{A}(\tau)\mathrm{d}\tau \\ &\qquad + \frac{1}{2}\left(\int_{t_0}^{t} \mathbf{A}(\tau)\mathrm{d}\tau\right)^2 \\ &\qquad + \frac{1}{3!}\left(\int_{t_0}^{t} \mathbf{A}(\tau)\mathrm{d}\tau\right)^3 + \cdots \Bigg)\mathbf{x}_0 \\ &= e^{\int_{t_0}^{t} \mathbf{A}(\tau)\mathrm{d}\tau}\mathbf{x}_0, \qquad (2.16)\end{aligned}$$

it is no longer at all obvious what the distribution of the process $\mathbf{x}(t)$ should be after marginalising over the latent force variables, or if indeed such a distribution exists. A more reasonable ambition is the existence of the various moments of the state variable and conditions for the existence of such solutions in the L_p sense are discussed in [Strand, 1970].

When these moments do exist it is worthwhile considering how they can be calculated precisely or at least approximated. Historically models of the form (2.10) have received some attention by physicists where, for example, they arise naturally in the optical Bloch equations [Arechhi and Bonifacio, 1965] —- looking to this literature for existing methods of solving this model in a way analogous to that done in Section 2.2.1. We note the existence of a matrix continued fraction method for deriving the marginal moments of the trajectory variables was presented in [Zoller et al., 1981, Dixit et al., 1980], see also [Risken, 1996], and differential equations for the marginal moments are given in [van Kampen, 1974] describing how the marginal moments evolve from some known initial condition. While the non-Gaussianity of the distribution means that, in general, we will not be able to recover the distribution from these moments, it is nevertheless of interest to review some of these methods here. They would, for example, allow us to approximate the unknown distribution with the moment matching Gaussian. However, because of the somewhat restrictive set of assumptions necessary for the following approximations we present only a brief review following the heuristic treatment in [van Kampen, 1974, 2007]. Under the same set of assumptions, these methods may be made rigorous, and we point the interested reader to the references as mentioned earlier. It will be useful to first rewrite the MLFM model (2.10) in the form

$$\frac{\mathrm{d}\mathbf{x}(t)}{\mathrm{d}t} = \mathbf{A}(t)\mathbf{x}(t) = [\mathbf{A}_0 + \alpha\mathbf{A}_1(t)]\mathbf{x}(t), \qquad \mathbf{x}(t_0) = \mathbf{x}_0 \tag{2.17}$$

where we have introduced the parameter α to measure the size of the random component, $\mathbf{A}_1(t)$, which we assume to be a mean zero, stationary stochastic processes. In the setting we are interested in this implies that each $\{g_r\}_{r=1}^{R}$ is a mean zero GP with stationary kernel function. Introducing the parameter α allows us to consider expansions in terms of the magnitude of the fluctuations of the random component, and in the limit $\alpha \to 0$ we recover a deterministic equation.

The most important, and ultimately restricting, assumption is that the random coefficient matrix possess a *finite correlation time* τ_c, that is for any pair of times t, t' we have that $|t - t'| > \tau_c$ implies that the random variables $\mathbf{A}(t)$ and $\mathbf{A}(t')$ are independent, we immediately note that this assumption is in general unlikely to be satisfied for popular choices of kernel function.

To approximate the solution of (2.17) it will be convenient to remove the offset matrix, $\mathbf{A}_0$, and we, therefore, introduce the variable $\mathbf{z}(t)$ defined by the transformation

$$\mathbf{z}(t) = e^{-\mathbf{A}_0 t}\mathbf{x}(t),$$

on differentiating, and assuming that in general $[\mathbf{A}_0, \mathbf{A}_1] \neq \mathbf{0}$, then we have

$$\begin{aligned}\frac{\mathrm{d}\mathbf{z}(t)}{\mathrm{d}t} &= \alpha e^{-t\mathbf{A}_0} A_1(t) e^{t\mathbf{A}_0}\mathbf{z}(t) \\ &\triangleq \alpha\mathbf{V}(t)\mathbf{z}(t).\end{aligned} \tag{2.18}$$

We also have that $\mathbf{z}(t_0) = \mathbf{x}(t_0)$. This enables us to rewrite the initial value problem (2.17) in the *interaction representation* form (2.18) which is in the form of (2.17) with $\mathbf{A}_0 = \mathbf{0}$. We can now derive a pathwise solution of (2.18) by way of the series expansion (2.15) for the solution of linear ODEs. For the current purposes it is enough to note that, up to terms of second order in the parameter α, the solution is approximately given by

$$\mathbf{z}(t) \approx \left(\mathbf{I} + \alpha\int_0^{t_1} V(\tau_1)\mathrm{d}\tau_1 + \alpha^2\int_0^{t}\int_0^{\tau_1} V(\tau_1)V(\tau_2)\mathrm{d}\tau_2\mathrm{d}\tau_1\right)\mathbf{x}_0, \tag{2.19}$$

Upon taking the expectation, and assuming that the initial condition is independent of $\mathbf{V}(t)$ then

$$\mathbb{E}\left[\mathbf{z}(t)\right] = \left(\mathbf{I} + \alpha^2\int_0^{t_1}\int_0^{t_2}\mathbb{E}\left[V(\tau_1)V(\tau_2)\right]\mathrm{d}\tau_1\mathrm{d}\tau_2\right)\mathbb{E}\left[\mathbf{x}_0\right]. \tag{2.20}$$

The mth term in the integral expansion (2.15) contributes a term of order $\mathcal{O}(\alpha^m t^m)$ and therefore for the approximation to be valid we would require that $\alpha t \ll 1$. Assuming this is the case and recalling that the process $\mathbf{A}_1(t)$ was assumed stationary we may rewrite (2.20) as

$$\mathbb{E}\left[\mathbf{z}(t)\right] = \left(\mathbf{I} + \alpha^2\int_0^{t}\int_0^{\tau_1}\mathbb{E}\left[\mathbf{V}(\tau_1)\mathbf{V}(\tau_1 - \tau_2)\right]\mathrm{d}\tau_2\mathrm{d}\tau_1\right)\mathbb{E}\left[\mathbf{x}_0\right]. \tag{2.21}$$

$$\mathbb{E}\left[\mathbf{z}(t)\right] = \left(\mathbf{I} + \alpha^2 \int_0^t \int_0^{\tau_1} \mathbb{E}\left[\mathbf{V}(\tau_1)\mathbf{V}(\tau_1 - \tau_2)\right] \mathrm{d}\tau_2 \mathrm{d}\tau_1 \right) \mathbb{E}\left[\mathbf{x}_0\right]. \tag{2.22}$$

The assumption that $\alpha t \ll 1$ is overly restrictive; our discussion of the control parameters in Chapter 1 made it clear that we would like to allow latent forces that exhibit quite significant fluctuations and so an apriori constraint that these forces only have large fluctuations for small times is undesirable. It can be shown [van Kampen, 1974] that under the finite correlation time assumption we can replace this small time restriction with the assumption that $\alpha \tau_c \ll 1$ allowing the integral in (2.22) to be extended to infinity leading to the approximation

$$\mathbb{E}\left[\mathbf{z}(t)\right] = \left(\mathbf{I} + \alpha^2 \int_0^t \int_0^{\infty} \mathbb{E}\left[\mathbf{V}(\tau_1)\mathbf{V}(\tau_2 - \tau_1) \mathrm{d}\tau_1 \mathrm{d}\tau_2\right] \right) \mathbb{E}[\mathbf{z}_0], \tag{2.23}$$

which we can identify as the solution to the linear differential equation

$$\frac{\mathrm{d}}{\mathrm{d}t}\mathbb{E}\left[\mathbf{z}(t)\right] = \alpha^2 \left[\int_0^{\infty} \mathbb{E}\left[\mathbf{V}(t)\mathbf{V}(\tau)\mathrm{d}\tau\right] \right] \mathbb{E}\left[\mathbf{z}(t)\right], \tag{2.24}$$

if we change back to the original variable then we derive an ODE for the expected value

$$\frac{\mathrm{d}}{\mathrm{d}t}\mathbb{E}\left[\mathbf{x}(t)\right] = \left[\mathbf{A}_0 + \alpha^2 \int_0^{\infty} \mathbb{E}\left[\mathbf{A}_1(t) e^{t\mathbf{A}_0} \mathbf{A}_1(t - \tau)\right] e^{-\tau \mathbf{A}_0} \mathrm{d}\tau \right] \mathbb{E}\left[\mathbf{x}(t)\right], \tag{2.25}$$

Approximations for higher order moments may be derived using a similar process. For example, if we define the product variables $u_{ij}(t) = x_i(t)x_j(t)$, for $i, j = 1, \ldots, K$ then

$$\begin{aligned} \frac{\mathrm{d}u_{ij}(t)}{\mathrm{d}t} &= \frac{\mathrm{d}x_i(t)}{\mathrm{d}t} x_j(t) + x_i(t) \frac{\mathrm{d}x_j(t)}{\mathrm{d}t} \\ &= \sum_{k=1}^{K} A_{ik}(t) u_{kj} + \sum_{k'=1}^{K} A_{jk} u_{ik'}, \end{aligned}$$

which is again a system of the form (2.17), the moments of which can be approximated using the same methods. If instead we wish to calculate the covariance then we will have to consider delay terms of the form $x_i(t)x_j(t - \tau)$, leading to a delay differential equation (DDE) version of the MLFM which would need to be solved for every time-lag in our observed dataset so that already these methods are of diminished practical utility.

For our purposes these methods are of limited for several reasons

(i) They focus on recovering the marginal distribution, rather than the conditional which allows us to classify motions with different controls.

(ii) They only return moments of the estimating distributions, rather than an approximation to the distributions themselves.

(iii) They require the finite correlation time and related assumptions which are unlikely to be satisfied.

Of these three caveats, the first is not necessarily a negative; indeed the possibility of constructing the distribution in the LFM after marginalising over the latent GPs allowed for the values of the remaining structural parameters to be learned more efficiently without reference to the latent GPs. This would be equally advantageous for the

MLFM also, as long as we also were able to recover the conditional distribution when it was required.

The second of the above points is more problematic, however, knowledge of the moments would at least allow us to consider useful approximations to the unknown distribution.

Ultimately it is the final point that is the most serious, in the setting we envision there is no reason to believe in a finite correlation time for the control processes. In Chapter 7 we consider applying our MLFM to motion capture data, and in this setting the control represent the inputs necessary to achieve a particular motion, and it seems likely that such processes are going to have a 'memory' that lasts over the whole interval during which the motion is carried out.

2.3.2 Latent force modelling and product manifolds

In the LFM of [Alvarez et al., 2009] the only permitted interactions between the state variables, $x_k(t)$, are through the common latent force variables, and the sensitivity matrix $\mathbf{S}$ governs the topology of these interactions. For example the conditions $S_{rk}S_{rl} = 0$ for all $r = 1, \ldots, R$ in (2.9) will lead to independence of the processes $x_k(t)$ and $x_l(t)$. Regardless of the entries of the sensitivity matrix, the process will ultimately be supported on a vector space of dimension less than or equal to K.

In contrast, the MLFM (2.10) will, in general, contain interactions between the state variables for non-trivial Lie algebras. A key feature of the MLFM extension is the embedding of geometric considerations, and therefore unlike the LFM, we do not consider the state space as being a Cartesian product of one-dimensional spaces. Instead we shall consider the full state space $\mathcal{X} \subset \mathbb{R}^K$ as a Cartesian product of Q submanifolds so that we have $\mathcal{X} = \mathcal{X}_1 \times \cdots \times \mathcal{X}_Q$ where each submanifold $\mathcal{X}_q$ is a manifold that is invariant under a particular choice of basis matrices $\{\mathbf{L}_d^{(q)}\}_{d=1}^{D_q}$, for $q = 1, \ldots, Q$, i.e. each $\mathcal{X}_q$ is a matrix Lie group with Lie algebra generated by the set $\{\mathbf{L}_d^{(q)}\}_{d=1}^{D_q}$ and contained in the space $\mathbb{R}^{K_q}$. The model is then specified in a hierarchical manner by conditioning on a smaller set of latent forces which will be shared by the trajectories on each of the submanifolds, that is

$$p(\mathbf{x} \mid \mathbf{g}) = \prod_{q=1}^{Q} p(\mathbf{x}^{(q)} \mid \mathbf{g}), \qquad \mathbf{x}^{(q)} \in \mathcal{X}_q \subset \mathbb{R}^{K_q}, \qquad \sum_{q=1}^{K} K_q = K,$$

where as previously $\mathbf{g}$ represents a set of R independent Gaussian processes.

This model structure allows for a flexible means of combining processes with values possibly on different manifolds, while still allowing them to share information through the common latent variables, a graphical representation of this construction is displayed in Figure 2.3. An analogous interpretation as a product manifold is available for the LFM, but in this case, the product topology is just the Cartesian product of one-dimensional real spaces.

Throughout this thesis we assume the geometry-dependent decomposition is known a priori, for example in Chapter 7 when we consider motion capture data the full dataset can naturally be valued as the product of rotation groups for each joint. However, it may be of interest to relax this assumption and allow for the manifold decomposition to instead be part of the learning process. In general this is a hard task but the approximate methods we consider in this thesis, and in particular, the resulting conditional independence structure leads to properties which make this a more realistic goal and we return to a discussion of this in the final chapter.

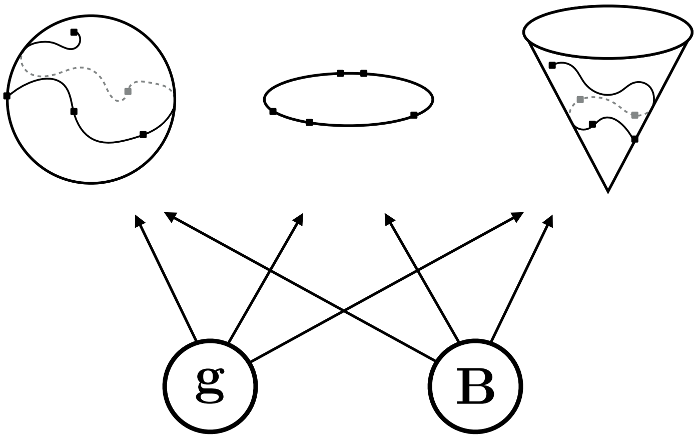


Figure 2.3: Representation of the MLFM as a decomposition of the process into trajectories on distinct submanifolds $\{\mathcal{X}^{(q)}\}_{q=1}^{Q}$ with shared latent forces. These manifolds each have a specific set of basis matrices $\{\mathbf{L}_d^{(q)}\}_{d=1}^{D_q}$, but share common trajectories modulated by the submanifold specific connection coefficients, $\mathbf{B}^{(q)}$.

2.4 Controllability and identifiability

Since linear ODEs are among the simplest class of models of dynamic systems it is not surprising that mathematically similar models to the LFM, and the multiplicative extension introduced in this chapter, have been well studied. Although less work has been done with the assumption that GPs model the forcing functions. Of particular relevance to our work is the field of control theory, [Songtag, 1998, Dorf and Bishop, 2011], in which (2.4) is an important mathematical model of a controlled dynamic system. In this setting the LFM described in Section 2.2 would typically be referred to as a linear time-invariant (LTI) control model, and the set of functions $\{g(t)\}_{r=1}^{R}$ are referred to as the *control(s)* of the system. The reproducing kernel Hilbert space associated with the latent forces represents the space of possible controls of this system, typically referred to as the admissable controls. Our extension with multiplicative interactions (2.10) has also been studied under the umbrella of control theory where it was introduced in [Jurdjevic and Sussmann, 1972] as the natural extension of the linear time-invariant control model to the Lie group setting.

Given the apparent connection with control systems we provide a brief review of the problems typically considered in the literature with a focus on how these directly relate to the problem we will be considering the remained of this thesis. For a particular set of parameters $\mathbf{A}$ and $\mathbf{S}$ defining an LTI system, equivalently the LFM, then one is typically interested in whether a system is *controllable* and *observable*.

For a dynamical system defined on an interval $[t_0, t_1]$ a state, $\mathbf{x}(t_0)$, is said to be *controllable*, [Kalman, 1960a], if there exists an admissible control $\mathbf{g}(t)$ defined on the interval such that $\mathbf{x}(t_1) = \mathbf{0}$. That is there exists a control which will transfer the initial state to the origin in some finite length of time; in general, the length of this interval will depend on the initial state. If this is true for all possible initial states, then the system is said to be *completely controllable.*

Similarly a linear system is said to be observable at time t_0 if $\mathbf{x}(t_0)$ can be determined from the output function, $\mathbf{y}(t)$, during some interval $[t_0, t_1]$, and if this is true for all t_0 and $\mathbf{x}(t_0)$, the system is said to be *completely observable.* In this thesis we shall typically consider the case where the output is given by a, noisy, observation of the state or in the case of the LFM exactly equal to the state so that observability is of less relevance here.

Controllability, then seeks to provide apriori guarantees that the structure of a model will lead to a controllable system, so that there is some function under the prior that that could lead to any observed trajectory, and from that perspective would seem to be a desirable guarantee to have for the LFM. The diagonal structure of the LFM means that this model is controllable so long as the entries on the diagonal are distinct [Kalman, 1963]. The Lie group setting is harder, however positive results have been established in [Sussmann and Jurdjevic, 1972] for the case when G is a compact Lie group, and the set of basis matrices $\{\mathbf{L}_d\}_{d=1}^{D}$ generates the Lie algebra $\mathfrak{g}$. This is the case for all of the examples we consider in this thesis.

A system structure parameterised by θ is said to be *globally identifiable* if for any two parameter vectors θ_1 and θ_2 in the parameter space the solution $\mathbf{x}(t, \theta) = \mathbf{x}(t\theta_2)$ for all $t \in [t_0, t_1]$ if and only if $\theta_1 = \theta_2$. Conditions for global identifiability in the LTI system, equivalently the LFM, are discussed in [Bellman and Åström, 1970]. Establishing identifiability for the MLFM would appear to be a much a harder problem.

If we restrict ourselves to a collection of linearly independent basis matrices then the problem would become easier, but with a significantly reduced flexibility. In our introduction of the MLFM we introduced the parameters β_{rd} and we remarked that in general the pair $\{\mathbf{g}, \mathbf{B}\}$ will not be identifiable. This is easily seen for the following simple complex valued version of the MLFM which we will discuss in more detail in Chapter 6

$$\frac{\mathrm{d}x(t)}{\mathrm{d}t} = i[\beta_0 + \beta_1 g(t)]x(t),$$

which is clearly equivalent to the system

$$\frac{\mathrm{d}x(t)}{\mathrm{d}t} = i[\tilde{g}(t)]x(t),$$

where $\tilde{g}(t) = \beta_0 + \beta_1 g(t)$. It follows that in the first specification there are redundant parameters which will not be identifiable. Clearly the second parameterisation is more parsimonious, but even so there are still good reasons to consider the first specification, for example, it makes clear the possibility of allowing the model to “switch off” the latent force during the fitting processes if $\beta_1 \to 0$ and so leading to a model with constant rate of rotation. This can be achieved in the second specification — but would typically require the kernel hyperparameters to tend towards degenerate limiting values which can lead to numerical issues. It is much more numerically stable to allow $\beta_1 \to 0$, than to achieve $g(t) \to c_0$ for some constant c_0 during the fitting process.

Furthermore, in a Bayesian setting the importance of parameter identifiability is less clear; if two distinct parameterisations realise identical dynamics with similar posterior probabilities, then this is not necessarily a shortcoming of the method. If we have some apparent reason to prefer one particular parameterisation over a distinct parameterisation realising the same dynamics, then this is information that should have been included when specifying the priors. Even in more typically frequentist settings, the importance of parameter identifiability is increasingly less clear with the advent of higher dimensional models. This is undoubtedly true for many modern machine learning methods, but even in more classical models such as regression with a large number of covariates the importance of parameter identifiability is diminished. Recent work by [Cox and Battey, 2017] considers the problem of classifying subsets of parameters that produce similar predictive distributions. Given that we are constructing hybrid models which do not place a meaningful interpretation on the mechanistic parameters we do generally take the view that prediction is more critical than identifiability for these models, and this is common stance in many modern statistical methods [Shmueli, 2010].

To summarise our stance in this thesis we do not discuss the identifiability, or controllability, of a particular model, and instead, we aim to introduce a framework which allows the specification and fitting of different models, and in particular the fitting of a maximal model. It is then up to the practitioner, if so desired, to assess the identifiability or controllability of the model they have chosen to fit either algebraically or using model diagnostics. If the model is not controllable, i.e. there is no combination of parameters and forces that would lead to an ODE with trajectory interpolation the observed data then this would quickly be revealed in the model fitting stage. The models introduced in this thesis typically include an observation noise parameter and higher point estimates, or distribution estimates with much mass around high values of the scale parameter, immediately suggest the chosen model has failed to interpolate the observed data well. In Chapter 6 we also discuss the “reconstruction error” at the MAP estimates of the model parameters, which we define to be the error between the observations and the result of solving the IVP using the MAP estimates of the parameters and latent forces, again a high value of this variable is indicative of the model not being controllable.

For prediction, our method is suitably flexible so that a maximal controllable model can be discovered, and it is then the goal of a model validation procedure combined with appropriate priors, to find a description of the system which displays good predictive performance. We demonstrate an example of this procedure in Chapter 7 using cross-validation to specify a parsimonious explanation of the model. With these remarks in mind we do not discuss identifiability of the point estimates we derive in this thesis, nevertheless establishing these properties, along with posterior consistency is something that would be a worthwhile area of study and we provide a few more remarks on this topic in Chapter 8.

2.5 Discussion

In this chapter, we have introduced the latent force model framework which we have defined to be the class of linear ODE models for which the time-dependent behaviour arises entirely through the variations of a set of smooth Gaussian processes. The LFM introduced by [Alvarez et al., 2009] and described in Section 2.3 can be considered from the point of view of control theory as a linear time-invariant control model with controls in the RKHS corresponding to a given set of kernel functions. Alternatively, from the statistics and machine learning perspective as a particular GP regression model with a non-stationary covariance function parameterised by the structure of the dynamical system of the model. Mathematically both of these descriptions are equivalent, but typically these fields differ in their objectives. In the control theory setting one usually introduces a control in order to achieve some desired behaviour. While from the statistical perspective one attempts to recover a control that realises the observed behaviour.

In Section 2.2.1 we noted that the fact that the LFM can be viewed as a particular case of GP regression makes it straightforward to carry out inference or use this model in more complex inferential structures. However, there are many situations in which the assumption of Gaussian trajectories implied by this model may be inappropriate. To allow for the construction of models which are not constrained to have Gaussian trajectories we have introduced an extension that allows for multiplicative interactions between the state and latent GP terms, and we have referred to this as the multiplicative latent force model.

Unlike the LFM, the MLFM can be constructed in such a way as to embed strong geometric constraints on the possible trajectories from the model. Unfortunately, this increase in modelling power comes at the expense of tractable inference. In section 2.3.1 we presented a series approximation to the solution for this model, as well as a class of approximations that can be realised from this series expansion, but require severely limiting assumptions. In the rest of this thesis we shall consider two methods of approximating the unknown distribution, a method in Chapter 3 which avoids explicitly solving the model and a second method in Chapter 4 constructed around the series expansion just discussed.

Finally, we concluded the section with some additional remarks on the LFM modelling setup. In Section 2.3.2 we discussed how to use the MLFM framework to construct very rich decompositions of a higher dimensional state variable as the cartesian product of lower dimensional manifolds. Information is then shared between these subsystems by allowing them to be driven by a common set of latent GPs. The LFM shares this decomposition structure, but in the linear case, the decomposition is less interesting. In this thesis the geometry of the data space is assumed known; however, we return to a discussion of this subject in Section 8.2.3 when we discuss the possibility of instead carrying out this decomposition in a lower dimensional latent variable space. In Section 2.4 we provided some brief remarks on the controllability and identifiability of these models, ultimately we do not provide a full analysis of the identifiability of any particular model, being more concerned with their predictive performance.

We discussed in the introduction of this thesis the conceptual utility of partition model parameters into structural and control parameters, and naturally, control theory provides a solid theoretical basis for the understanding of such systems. Nevertheless, while the models are mathematically equivalent, we do see a distinction between the application of these models in the statistical machine learning setting and their analysis from control theory. Often in control theory one begins with some desirable behaviour for a given physical system and then attempts to find a control that will achieve this behaviour, this process is referred to as optimal control theory [Kalman, 1960b]. On the other hand, we view the latent force modelling framework as allowing us to learn a possible control that will produce a given system, and then to classify and discuss qualitative aspects of this system through a discussion of the learned controls, rather than a discussion of the trajectories in state space. We provide an example of this process in Chapter 7.

In summary, the latent force modelling framework allows for the construction of dynamic systems driven by flexible GPs. The LFM is a Gaussian process, and so displays ideal properties for carrying out inference. In constructing our methods for approximate inference in the MLFM in the following chapters we shall be guided by the desire to stay as close to this idealistic scenario, and indeed in both methods, we shall demonstrate the possibility of introducing approximations that are conditionally Gaussian. As a result, we will be able to demonstrate that the increase in the modelling possibilities of the MLFM comes at only a small sacrifice in analytic tractability.

Chapter 3

Adaptive gradient matching

3.1 Introduction

In the previous chapter we discussed a class of dynamic systems in which the time-dependent behaviour of the, linear, evolution equation arose entirely through the variations of a set of independent smooth Gaussian processes. Our discussion centred on an extension of the LFM framework of [Alvarez et al., 2009] allowing for multiplicative interactions between the latent forces and the state variables, which we referred to as the *multiplicative latent force model* (MLFM). This extended model is contained within the broader class of non-autonomous linear ODEs, and one important feature of this class is that in general there is no closed form solution for the trajectory. In order to perform inference we would, ideally, possess a representation of the trajectory as some function of the initial conditions, and the latent forces, allowing us to investigate how the stochastic properties of the latent variables propagate through to the observed trajectories.

The lack of explicit pathwise solutions for the MLFM was in contrast to the case for the LFM for which the solutions took the simple form (2.5), and so allowed the trajectories to be expressed as a linear transform of the latent GPs. This important property of the LFM framework, which made subsequent inference straightforward, implies a joint Gaussian distribution for the state variables and the latent force variables so that the LFM may be viewed as a particular instance of the more general GP regression modelling paradigm.

These properties that made the LFM attractive from the point of view of tractable inference are at the same time one of the most significant limitations of the LFM model, and as such will be lost when considering our proposed extension to allow multiplicative interactions. Unfortunately any inference for the MLFM must be carried out in the absence of an explicit formula for the solution in terms of the latent forces, and so it is not immediately obvious what the resulting probabilistic structure of this model will be. Certainly the possibility of strong geometric constraints on the space of possible solutions strongly suggest it will no longer have the convenient structure of a GP regression model, and indeed we observe that, in extending the model to the class of multiplicative interactions, we will necessarily lose the attractive joint Gaussian distribution of the state and latent force variables. Much of the work in this thesis will be devoted to examining the problem of how to carry out inference for the MLFM, and to what extent the appealing properties of the LFM must be sacrificed for a greater control of the model geometry. Throughout this chapter, and those that follow, we will be guided by the desire to construct inferential methods that are accurate, but are also as close as possible to the idealistic scenario of the LFM in terms of computational efficiency and analytical tractability.

The methods that we shall consider in this chapter may be broadly referred to as adaptive gradient matching methods, and are constructed by modifying a class of existing methods for carrying out Bayesian inference for a finite dimensional parameter set in a very general class of, possibly nonlinear, ODE models without explicitly, or numerically, solving the ODE itself. We will show that on introduction of these approximations we replace the ideal joint Gaussian distribution of the forces and state variables in the LFM with a conditional Gaussian structure.

Gradient matching methods exploit the existence of the explicit parametric relationship between the state and its gradient, as embodied in the models evolution equation. The early approach in [Varah, 1982], and subsequent improvements in [Ramsay et al., 2007], used spline interpolants to obtain initial estimates of the trajectory and its gradient, and these interpolants were then used to provide estimates of the parameters. Our interest will be in further developments replacing the spline interpolators with the use of GP interpolants as carried out in [Calderhead et al., 2009] and the extension introduced in [Dondelinger et al., 2013]. The replacement of the spline approximations with GP trajectories gives the model a richer probabilistic structure which better enables the incorporation of unobserved data or misspecified problems, [Tarantola, 2005]. While the emphasis in these methods is primarily on well specified mechanistic models, where the model uncertainty is expressed by the stochasticity of a, usually small, set of parameters it still seems reasonable to expect that such methods should be applicable to the class of latent force models, linear ODEs being contained within the much wider class of general parametrised ODE models. It further seems plausible that the simpler structure of the linear case will result in an inferential process that is substantially simpler than the general case. In this chapter we shall show that this hoped for simplification does indeed occur.

Before we introduce the adaptive gradient matching methods we should mention an alternative strategy of constructing a generative method for this class of models by; first simulating the ODE parameters, then solving the given ODE using some numerical ODE method and then finally assessing the fit to the observed trajectory. An example of the application of this method to problems from systems biology is presented in [Vyshemirsky and Girolami, 2008]. This method is certainly practical for small parameter sets, and even in the case of infinite dimensional parameters such as Gaussian processes it is not beyond the realms of possibility. This method would seem like a perfectly viable option for producing simulated samples from random ODE models, however they provide no insight into what an analytic form, or an approximation thereof, for the resulting conditional might look like, and it is for this reason we do not discuss these methods in more detail. In this chapter we are able to construct the full probabilistic structure of an approximation to the MLFM, and this will leave us in a position to provide deterministic approximation methods in later chapters, something that would be out of reach with these simulation based approaches.

In the remainder of this chapter, we introduce the adaptive gradient matching processes in the most general setting for nonlinear ODEs. We describe the underlying probabilistic structure and the resulting form of the joint density of the state and model parameters. We then consider the particular case in which the evolution equation is that of our MLFM model demonstrating for the variables in which we are primarily interested it is possible to provide a complete set of conditional distributions. We shall see that the presence of tractable conditional posteriors allows for a simpler sampling scheme than that required for the nonlinear case, and closely related to this result will be the ability to derive variational approximations which we shall discuss in Chapter 5. Finally, we conclude with a discussion of the introduction of the Gaussian processes interpolant for the latent trajectory, and in particular the hyperparameters of the interpolant. Notable is that the resulting posterior of the latent forces necessarily depends not only on the observed values of the latent Gaussian forces but also on the hyperparameters and we discuss some of the implications of this feature when data is sparse relative to the model structure.

3.2 Bayesian adaptive gradient matching for ODEs

As we discussed in the previous chapter the greatest barrier to carrying out inference in the MLFM is the absence of a simple closed form for the fundamental solution. Of course this problem is even more apparent in the general case of nonlinear ODEs, where even a series expansion of the solution may not be possible. It is therefore interesting that despite this impediment there has been progress in the development of inference methods for dealing with this class of models. One particularly noteworthy example of these are the *gradient matching* approaches initially proposed in [Varah, 1982] and further developed in [Ramsay et al., 2007], for carrying out parameter inference in nonlinear ODEs. These methods exploit the fact that while we have no closed form expression for the trajectory in terms of any model parameters we do have an explicit expression for the gradient of the trajectory — it is precisely the model's evolution equation. Conditional then on an estimate of the trajectory, and its gradient, the relationship as expressed by the evolution equation allows for the possibility of introducing a measure of 'goodness-of-fit' of a particular parameter.

The complexity is in proposing a suitable estimate of the trajectory and its gradient on the basis of discrete observations of the state variable and then updating this estimate in a principled manner. The approaches of [Varah, 1982, Ramsay et al., 2007] is to construct these estimates of the state trajectories using spline interpolants of the observed data and then comparing the fidelity of the interpolated state and its gradient with that implied by the model equation, and then using this measure of fit to carry out inference. While these methods are successful for motivating point estimates for the parameters it is less clear how to carry out Bayesian inference or deal with missing data in the spline framework, and therefore we shall be particularly interested in the extension considered by [Calderhead et al., 2009] and further improved in [Dondelinger et al., 2013] where the estimates of the state and its gradient are given by Gaussian process interpolants so providing the model with probabilistic structure and allowing us to carry out fully Bayesian inference. In this section we review the Bayesian adaptive gradient matching methods in the most general setting before considering the restriction to linear models of the MLFM form in the following section where we demonstrate certain simplifications that occur in the linear case.

3.2.1 Model specification

We describe the general setup of the Bayesian adaptive gradient methods using the notation as described in [Calderhead et al., 2009, Dondelinger et al., 2013]. For the general case we are interested in dynamic models where a K-dimensional latent state variable $\mathbf{x}(t) \in \mathbb{R}^K$ evolves according to a parametrised, potentially nonlinear, ODE denoted by

$$\frac{\mathrm{d}\mathbf{x}(t)}{\mathrm{d}t} = f(\mathbf{x}; \boldsymbol{\theta}), \tag{3.1}$$

where the continuous time-evolution function $f(\cdot; \boldsymbol{\theta}) : \mathbb{R}^K \to \mathbb{R}^K$ depends on the random parameter $\boldsymbol{\theta}$. In later developments we shall be interested in the case where $\boldsymbol{\theta}$ is an infinite dimensional time indexed parameter but for now we can consider it as some arbitrary random vector for which we are able to specify a prior which we shall denote by $p(\boldsymbol{\theta})$. We will also denote by $f_k(\mathbf{x}; \boldsymbol{\theta})$ the scalar function returning the kth component of the vector valued function time-evolution function.

For carrying out posterior inference we will be interested in a set of T time points $t_1 < \cdots < t_T$ for which we have obtained a sequence of, possibly noisy, observations $\mathbf{Y} = \{\mathbf{y}(t_1), \ldots, \mathbf{y}(t_T)\}$ of the dynamic system at each of these time points. Each $\mathbf{y}(t_i)$ is assumed to be an independent noisy observation of the latent variable $\mathbf{x}(t_i)$ whose evolution is described by (3.1). We specify an additive noise model relating these quantities of the form

$$\mathbf{y}(t) = \mathbf{x}(t) + \epsilon(t),$$

where $\epsilon(t)$ is a K-dimensional multivariate Gaussian noise vector with mean zero. We shall assume that the error terms also have a diagonal covariance $\mathbb{E}\left[\epsilon_i(t)\epsilon_j(t)\right] = \delta_{ij}\sigma_i^2$, for $i, j = 1, \ldots, K$, where δ_{ij} is the Kronecker delta with $\delta_{ij} = 1$ if $i = j$ and zero otherwise. The error distribution is therefore completely parameterised by the K dimensional vecotr $\boldsymbol{\sigma} = (\sigma_1, \ldots, \sigma_K)^\top$. We also denote by $\mathbf{X}$ the complete sequence of latent vectors for our given time set T, i.e. $\mathbf{X} = \{\mathbf{x}(t_1), \ldots, \mathbf{x}(t_T)\}$. Having specified the dynamics it follows that after solving for the locations $\mathbf{x}(t) = \mathbf{x}(t; \boldsymbol{\theta})$ that the observed data $\mathbf{Y}$ has conditional density

$$\begin{aligned} p(\mathbf{Y} \mid \mathbf{X}, \boldsymbol{\sigma}) &= \prod_{k=1}^{K} \prod_{t=1}^{T} p(y_k(t) \mid x_k(t), \sigma_k) \\ &= \prod_{k=1}^{K} \prod_{t=1}^{T} \mathcal{N}(y_k(t) \mid x_k(t), \sigma_k^2). \end{aligned} \tag{3.2}$$

We use the notation $\mathbf{x}_k$ to denote the vector in $\mathbb{R}^T$ given by the time series $(x_k(t_1), \ldots, x_k(t_T))^\top$, and similarly for $\mathbf{y}_k$. We shall also use the bold font lower case $\mathbf{x}$ and $\mathbf{y}$ without subscript to denote the NK dimensional vectors $\mathbf{x} = (\mathbf{x}_1, \ldots, \mathbf{x}_K)^\top$, and the uppercase $\mathbf{X}, \mathbf{Y}$ to refer to the complete set of these random variables without reference to shape. The vector random variable $\mathbf{x}$ and the collection $\mathbf{X}$ are identified with one another in the obvious way, when referring to their density functions we shall typically use the uppercase for generic representations of the density, such as $p(\mathbf{X})$, and the lower case for commonly appearing density functions, such as the Gaussian $\mathcal{N}(\mathbf{x} \mid \boldsymbol{\mu}, \boldsymbol{\Sigma})$, which are defined over vector spaces.

Having specified the prior distribution on the parameter as well as the observation noise model it remains to specify the likelihood term of the latent states $p(\mathbf{X} \mid \boldsymbol{\theta})$. As we have discussed this will be difficult in general because the trajectory is defined only implicitly by the initial state and parameters through the differential equation (3.1). Only in a few special cases will a closed form solution for the trajectory exist — one of these is the constant coefficient linear ODE with inhomogeneous forcing term that we discussed in the previous chapter. To circumvent this problem [Calderhead et al., 2009] first place an independent mean zero Gaussian process prior on each of the K components of the trajectory *independently of the model parameters.* We denote each of these priors by

$$p(\mathbf{x}_k \mid \boldsymbol{\phi}_k) = \mathcal{N}(\mathbf{x}_k \mid \mathbf{C}_{\boldsymbol{\phi}_k}), \tag{3.3}$$

where $\mathbf{C}_{\boldsymbol{\phi}_k}$ denotes the covariance matrix with entities given by a kernel function parametrised by $\boldsymbol{\phi}_k$. Combining this with the assumption of independent additive noise one can marginalise out the latent states, $\mathbf{x}_k$, to give the Gaussian process regression posterior for $\mathbf{y}_k$ given by

$$\begin{aligned} p(\mathbf{y}_k \mid \boldsymbol{\phi}_k, \sigma_k) &= \int p(\mathbf{y}_k \mid \mathbf{x}_k, \sigma_k) p(\mathbf{x}_k | \boldsymbol{\phi}_k) \mathrm{d}\mathbf{x}_k \\ &= \int \mathcal{N}(\mathbf{y}_k \mid \mathbf{x}_k, \sigma_k^2 \mathbf{I}_T) \mathcal{N}(\mathbf{x}_k \mid \mathbf{0}, \mathbf{C}_{\boldsymbol{\phi}_k}) \mathrm{d}\mathbf{x}_k \\ &= \mathcal{N}(\mathbf{y}_k \mid \mathbf{0}, \mathbf{C}_{\boldsymbol{\phi}_k} + \sigma_k^2 \mathbf{I}_T), \end{aligned} \tag{3.4}$$

where $\mathbf{I}_T$ is the $T \times T$ identity matrix. At this point it should be emphasised that no reference has been made to the model parameters, this is a likelihood term for the data $\mathbf{y}$ made in ignorance of the differential equation structure of the model, rather than as the likelihood with the ODE structure integrated out.

Since the model in its current form is merely a standard GP regression and in order to carry out inference for the ODE model, and the related parameters, we now need to provide a link between the parameters $\boldsymbol{\theta}$ and the trajectories $\mathbf{x}(t)$ which avoids explicitly solving the model. The important observation underlying adaptive gradient methods is the access to an explicit, and often simple, expression for the state gradients as a known transformation of the state variables and the model parameters given by the specified time-evolution equation (3.1). The Bayesian adaptive gradient matching approaches propose to introduce the state gradients into the model, and then attempt to marginalise them out in such a way that the resulting density is tractable. Since we have assumed that each of the Gaussian processes priors is almost surely differentiable then we can conclude that the state derivatives exist and will themselves be Gaussian processes, [Solak et al., 2003]. At this point we introduce the compact notation $\dot{\mathbf{x}}(t) := \mathrm{d}\mathbf{x}(t)/\mathrm{d}t$, for the derivative of a process with respect to time, allowing us to compactly write the conditional distribution of $\dot{\mathbf{x}}_k := \{\dot{\mathbf{x}}(t_1), \ldots, \dot{\mathbf{x}}(t_T)\}$ as

$$p(\dot{\mathbf{x}}_k \mid \mathbf{x}_k, \boldsymbol{\phi}_k) = \mathcal{N}(\mathbf{x}_k \mid \mathbf{m}_{\dot{x}_k|x_k}, \mathbf{C}_{\dot{x}_k|x_k}). \tag{3.5}$$

The mean and covariance matrices of this conditional distribution are given by

$$\mathbf{m}_{\dot{x}_k|x_k} = \mathbf{C}_{x_k\dot{x}_k}^{\top} \mathbf{C}_{\phi_k}^{-1} \mathbf{x}_k \tag{3.6a}$$

$$\mathbf{C}_{\dot{x}_k|x_k} = \mathbf{C}_{\dot{x}_k\dot{x}_k} - \mathbf{C}_{x_k\dot{x}_k}^{\top} \mathbf{C}_{\phi_k}^{-1} \mathbf{C}_{x_k\dot{x}_k}, \tag{3.6b}$$

where we have defined the matrices $\mathbf{C}_{x_k\dot{x}_k} = \mathbf{C}_{x_k\dot{x}_k}(\boldsymbol{\phi}_k)$ which represents the cross covariance between the state and its derivatives, and $\mathbf{C}_{\dot{x}_k\dot{x}_k} = \mathbf{C}_{\dot{x}_k\dot{x}_k}(\boldsymbol{\phi}_k)$ which denotes the covariance of the gradient process. These matrices will have entries given by

$$[\mathbf{C}_{x_k\dot{x}_k}]_{ij} = \left.\frac{\partial k(s,t;\boldsymbol{\phi}_k)}{\partial t}\right|_{s=t_i,\, t=t_j}, \qquad [\mathbf{C}_{\dot{x}_k\dot{x}_k}]_{ij} = \left.\frac{\partial^2 k(s,t;\boldsymbol{\phi}_k)}{\partial s \partial t}\right|_{s=t_i,\, t=t_j}, \tag{3.7}$$

for $i,j = 1,\ldots,T$. We shall also denote the $T \times T$ matrix $\mathbf{C}_{\dot{x}_k|x_k}\mathbf{C}_{x_k}^{-1}$ by $\mathbf{M}_k$ which acts to transform the state vector $\mathbf{x}_k$ to the conditional mean of the gradient under the GP prior. This gives the distribution of the state gradients as implied by the prior, however we have not as yet introduced any of the information arising from the ODE model specification (3.1). This is done in [Calderhead et al., 2009] by considering a separate conditional distribution involving an additive noise term with variance γ_k for each component and defining the nonlinear regression model

$$p(\dot{\mathbf{x}}_k \mid \mathbf{X}, \boldsymbol{\theta}, \gamma_k) = \mathcal{N}(\dot{\mathbf{x}}_k \mid \mathbf{f}_k(\mathbf{X}, \boldsymbol{\theta}), \gamma_k \mathbf{I}), \tag{3.8}$$

where the N-vector $\mathbf{f}_k$ has entries given by $\mathbf{f}_{ki} = f_k(\mathbf{x}(t_i), \boldsymbol{\theta})$. This conditional distribution then acts to centre the gradient on the values given by the model evolution equation (3.1) with flexibility coming from the parameters γ_k.

As it stands, the Gaussian process approximation of the state and its gradient are disjoint from the estimate arising from the nonlinear regression model (3.8) and so both conditional distributions must be combined in an appropriate way. The approach taken in [Calderhead et al., 2009] is to combine the conditional density arising from the prior (3.5), with (3.8) by using a multiplicative product of experts approximation, [Hinton, 2002], and then integrate over the gradient variables. That is we form the approximate distribution

$$\begin{aligned} p(\dot{\mathbf{x}}_k \mid \mathbf{X}, \boldsymbol{\theta}, \boldsymbol{\phi}, \boldsymbol{\gamma}) &\propto p(\dot{\mathbf{x}}_k \mid \mathbf{x}_k, \boldsymbol{\phi}_k) \times p(\dot{\mathbf{x}}_k \mid \mathbf{X}, \boldsymbol{\theta}, \boldsymbol{\gamma}) \\ &\propto \mathcal{N}(\dot{\mathbf{x}}_k \mid \mathbf{m}_{\dot{x}_k|x_k}, \mathbf{C}_{\dot{x}|x}) \times \mathcal{N}(\dot{\mathbf{x}}_k \mid \mathbf{f}_k(\mathbf{X}, \boldsymbol{\theta}), \gamma_k \mathbf{I}). \end{aligned} \tag{3.9}$$

The first component in (3.9) is the component arising from the prior model, and the second is that arising from the regression model using the evolution equation. The result is a conditional density which places most of its mass around estimates of the gradient which agree with the parametrised model (3.8), but also coincide with the gradient of a GP interpolant. This product of experts density can now be used to marginalise out the gradients using

$$\begin{aligned} p(\mathbf{X} \mid \boldsymbol{\theta}, \boldsymbol{\phi}, \boldsymbol{\gamma}) &= \int p(\mathbf{X}, \dot{\mathbf{X}} \mid \boldsymbol{\phi}) \mathrm{d}\dot{\mathbf{X}} \\ &= \prod_{k=1}^{K} \int p(\dot{\mathbf{x}}_k \mid \mathbf{x}_k, \boldsymbol{\phi}_k) p(\dot{\mathbf{x}}_k \mid \mathbf{X}, \boldsymbol{\theta}, \gamma_k) \mathrm{d}\dot{\mathbf{x}}_k \; p(\mathbf{X} \mid \boldsymbol{\phi}) \\ &= \prod_{k=1}^{K} \int \mathcal{N}(\dot{\mathbf{x}}_k | \mathbf{m}_k, \mathbf{C}_{\dot{x}_k|x_k}) \mathcal{N}(\dot{\mathbf{x}}_k \mid \mathbf{f}_k(\mathbf{X}, \boldsymbol{\theta}), \gamma_k^2 \mathbf{I}) \mathrm{d}\dot{\mathbf{x}}_k \; p(\mathbf{X} \mid \boldsymbol{\phi}) \\ &\propto \frac{1}{\prod_{k=1}^{K} Z(\gamma)_k} \exp\left\{ -\frac{1}{2} \mathbf{x}_k^{\top} \mathbf{C}_{\phi_k} \mathbf{x}_k - \frac{1}{2} \sum_{k=1}^{K} (\mathbf{f}_k - \mathbf{m}_k)^T (\mathbf{C}_{\dot{x}_k|x_k} + \gamma_k^2 \mathbf{I})^{-1} (\mathbf{f}_k - \mathbf{m}_k) \right\}, \end{aligned} \tag{3.10}$$

where $Z(\gamma)_k = |2\pi(\mathbf{C}_{\dot{x}_k|x_k} + \gamma_k^2 \mathbf{I})|^{1/2}$. This is now sufficient to give a joint density over the state variables and model parameters.

The process of introducing two different conditional densities, one from the prior and one from the ODE model, and then collapsing them using a product of experts approach is displayed graphically in Figure 3.1. This approach to model building has a necessary dissonance between the two "experts", for example it is easy to consider model specifications where the trajectories will certainly not have a Euclidean support. This non-Euclidean support immediately violates the GP prior represented by the first expert so that these two experts motivate beliefs that are not consistent with one another.

3.2.2 Parameter inference

Leaving aside questions concerning the accuracy of the product of experts approximations, and the philosophical implications surrounding it, we still need to consider how to use this approximation to carry out inference for the complete model. The full joint density of the model can be obtained by combining the gradient matching approximation with priors on the model parameters, $\boldsymbol{\theta}$, the GP hyperparameters $\boldsymbol{\phi}$, the gradient matching regularisation parameters $\boldsymbol{\gamma}$, and finally the observation noise parameter, $\boldsymbol{\sigma}$. We write

$$\begin{aligned} p(\mathbf{Y}, \mathbf{X}, \boldsymbol{\theta}, \boldsymbol{\phi}, \boldsymbol{\sigma}, \boldsymbol{\gamma}) \\ &= p(\mathbf{Y} \mid \mathbf{X}, \boldsymbol{\sigma}) p(\mathbf{X} \mid \boldsymbol{\theta}, \boldsymbol{\phi}, \boldsymbol{\gamma}) p(\boldsymbol{\sigma}) p(\boldsymbol{\theta}) p(\boldsymbol{\phi}) p(\boldsymbol{\gamma}), \end{aligned} \tag{3.11}$$

where the conditional $p(\mathbf{X} \mid \boldsymbol{\theta}, \boldsymbol{\phi}, \boldsymbol{\gamma})$ is given by (3.10). However, because of the complex, and potentially nonlinear, dependence of the vectors in $\mathbf{f}_k$ on the state variables it will, in general, not be possible marginalise (3.11) and so provide a closed form conditional posterior.

An initial approach to handling the intractability of the joint density suggested by [Calderhead et al., 2009] was to use a collapsed Gibbs sampling procedure with the following partition of the complete set of models parameters

$$\boldsymbol{\phi}, \boldsymbol{\sigma} \sim p(\boldsymbol{\phi}, \boldsymbol{\sigma} \mid \mathbf{Y}) \tag{3.12a}$$

$$\mathbf{X} \sim p(\mathbf{X} \mid \mathbf{Y}, \boldsymbol{\sigma}, \boldsymbol{\phi}) \tag{3.12b}$$

$$\boldsymbol{\theta}, \boldsymbol{\gamma} \sim p(\boldsymbol{\theta}, \boldsymbol{\gamma} \mid \mathbf{X}, \boldsymbol{\phi}). \tag{3.12c}$$

The first of these sampling steps, (3.12a), involves sampling from the distribution with density given, up to a normalisation constant, by a marginalisation of (3.11). In particular we have

$$\begin{aligned} p(\boldsymbol{\phi}, \boldsymbol{\sigma} \mid \mathbf{Y}) &\propto p(\boldsymbol{\phi}) p(\boldsymbol{\sigma}) \\ &\times \int\int\int p(\mathbf{Y} \mid \mathbf{X}, \boldsymbol{\sigma}) p(\mathbf{X} \mid \boldsymbol{\theta}, \boldsymbol{\phi}, \boldsymbol{\gamma}) p(\boldsymbol{\theta}) p(\boldsymbol{\gamma}) \mathrm{d}\boldsymbol{\theta} \mathrm{d}\boldsymbol{\gamma} \mathrm{d}\mathbf{X}. \end{aligned} \tag{3.13}$$

Similarly the density in (3.12b) is formally obtained by normalising the expression

$$\begin{aligned} p(\mathbf{X} \mid \mathbf{Y}, \boldsymbol{\sigma}) &\propto \\ &\int\int\int p(\mathbf{Y} \mid \mathbf{X}, \boldsymbol{\sigma}) p(\mathbf{X} \mid \boldsymbol{\theta}, \boldsymbol{\phi}, \boldsymbol{\gamma}) p(\boldsymbol{\gamma}) p(\boldsymbol{\phi}) p(\boldsymbol{\theta}) \mathrm{d}\boldsymbol{\theta} \mathrm{d}\boldsymbol{\phi} \mathrm{d}\boldsymbol{\gamma}, \end{aligned} \tag{3.14}$$

and finally we have

$$p(\boldsymbol{\theta}, \boldsymbol{\gamma} \mid \mathbf{X}, \boldsymbol{\phi}) \propto p(\boldsymbol{\theta}) p(\boldsymbol{\gamma}) p(\mathbf{X} \mid \boldsymbol{\theta}, \boldsymbol{\gamma}, \boldsymbol{\phi}). \tag{3.15}$$

We note that all of the above marginal conditionals make use of the product of experts approximation (3.10), and so in general the integrands in (3.13) and (3.14) will be intractable.

Given the intractability of the sampling scheme (3.12), [Calderhead et al., 2009] proposed to instead sample from an approximation to this scheme given by

$$\boldsymbol{\phi}, \boldsymbol{\sigma} \sim p^*(\boldsymbol{\phi}, \boldsymbol{\sigma} \mid \mathbf{Y}) \propto \int p(\mathbf{Y} \mid \mathbf{X}) p_{GP}(\mathbf{X} \mid \boldsymbol{\phi}) \tag{3.16a}$$

$$\mathbf{X} \sim p^*(\mathbf{X} \mid \mathbf{Y}, \boldsymbol{\sigma}) \propto p(\mathbf{Y} \mid \mathbf{X}, \boldsymbol{\sigma}) p_{GP}(\mathbf{X} \mid \boldsymbol{\phi}) \tag{3.16b}$$

$$\boldsymbol{\theta}, \gamma \sim p(\boldsymbol{\theta}) p(\gamma) p(\mathbf{X} \mid \boldsymbol{\theta}, \gamma, \boldsymbol{\phi}), \tag{3.16c}$$

where $p_{GP}(\mathbf{X} \mid \boldsymbol{\phi})$ is the prior GP regression model given by (3.3), and (3.16c) is again given by the product of experts approximation.

By sampling now from (3.16a) and (3.16b), one avoids the intractable integrals which are present in (3.12), infact (3.16b) is available in closed form using standard properties of conditional Gaussians given in Appendix A.2. Furthermore, while the full distribution (3.16a) is not typically available in closed form because of the the nonlinear dependence of the GP kernel function on the hyperparameters, the integral nevertheless is tractable allowing a more potentially more efficient sampling scheme by collapsing the state variable.

However, while the approximate scheme (3.16) is computationally much more efficient it possesses a significant conceptual flaw as noted by [Dondelinger et al., 2013]. In particular the collapsed approximation (3.16a) combined with the update of the state variable using (3.16b) means that the update of the state interpolating GP model is only ever done using information from the prior model. In effect, a GP is fitted to the data first, without knowledge of the system dynamics, and the parameters are subsequently inferred from this interpolant, leading to a two step process with no mechanism for the parameters to inform the interpolant. The motivation for this approximation goes that these variables, and particularly the kernel hyperparameters, are nuisance variables, and so we should prefer a cheaper update so long as this does not have a negative impact on our inference of the more critical variables. Numerical evidence for the possible poor performance of the scheme (3.16) was presented in [Dondelinger et al., 2013], who instead propose to estimate all the parameters by sampling from the full complete joint density (3.11) by using a Metropolis-Hastings (MH) [Hastings, 1970] update. They demonstrate that the computational complexity of this method with regards to run time is acceptable compared to the original gradient matching method, and more importantly that the method is able to provide accurate parameter inference in situations where the original method fails to converge. Because the hyperparameters are now coupled to both the system parameters and state variables through the complete joint, we can view the interpolant as now being regularised by the underlying dynamical system, consequently [Dondelinger et al., 2013] refer to the resulting method sampling from the full joint without marginalisation as *adaptive gradient matching*, to indicate the existence of the feedback mechanism between the interpolant hyperparameters, and the model parameters which encode the dynamical systems information. We shall see in the next section that in the case of linear ODEs we can both follow [Dondelinger et al., 2013] in sampling from the proper density (3.11), but that it is also possible to partition the variables in such a way the MH sampling may be replaced by Gibbs sampling.

At this point, it is worth considering sampling approaches that we might use if we were instead willing to solve the ODE directly. An important class of such generative simulation-based methods are the Approximate Bayesian Computation (ABC) approaches to solving this problem [Tavaré et al., 1997, Beaumont et al., 2002]. Directly solving the ODE using numerical methods necessitates introducing a step-size parameter and solving the ODE over a dense set of time points. Furthermore, in the

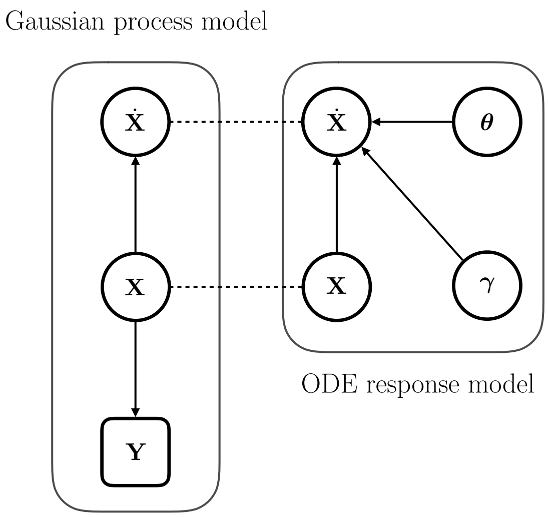


Figure 3.1: Representation of the product of experts assumption in the adaptive gradient matching method of Calderhead et al. [2009]. The lefthand panel represents a conventional Gaussian process regression model, while the right hand encodes the conditional density of the state variables given the model parameters and the tuning parameter $\boldsymbol{\gamma}$. These are combined as a multiplicative product by identifying the state variables connected by the '- - -' line using the product of experts approximation (3.9).

setting of the MLFM, we must also consider a suitably dense realisation of the latent forcing functions for each simulation of the trajectory. Even if such an approach is feasible, there are still good reasons to prefer the approximate method discussed in this chapter. The first reason to prefer the sampling approach described in this section to an ABC-type approach concerns the practical implementation, the set of latent forces which solve the ODE through the set of observed data may occupy a very narrow region of the infinite dimensional function space leading to difficulties in constructing a suitable sampling scheme. At the very least it is likely to require a very well specified initial guess of the latent force. Compare this with the method in this chapter where a good initial guess of the latent states is much simpler to construct; indeed the GP prior is a reasonable choice and is the choice already discussed for the sampling scheme (3.16b). After combining with a reasonable initial guess for the finite dimensional connection coefficient parameters for we can begin drawing samples from the (approximate) posterior immediately. By comparison, it is much harder to propose a suitable initial value for ABC, an arbitrary sample from the prior is very unlikely to solve the ODE through, or even close to, the observed data. The second benefit is that by providing an analytic form for the conditional posteriors it will be much easier to construct deterministic approximations to the posterior as well as sampling-based methods, and we demonstrate this feature in Chapter 5.

In summary, the use of adaptive gradient matching methods will allow us to carry out approximate sampling from a very general class of nonlinear ODE models without ever explicitly solving the ODE, and we now turn to consider the explicit form this approximation takes for the MLFM.

3.3 Adaptive gradient matching for the MLFM

Having now described, in full generality, the Bayesian adaptive gradient matching method for carrying out parameter inference for nonlinear ODEs with random parameters, we now consider how to adapt this model so that it may be applied to the MLFM introduced in Chapter 2. The adaptive gradient matching methods were designed to be applied to nonlinear ODEs with a, typically small, set of parameters while we are interested in a linear ODE with an infinite dimensional parameter vector. Nevertheless, it seems reasonable to expect that if we view linear ODEs as a subclass of the family of nonlinear ODE models and an infinite dimensional parameter as just a very large, but finite, dimensional parameter that we could expect the methods introduced in the previous section to equally apply to the MLFM. Furthermore, we might expect that the original inference scheme will possess some attractive simplifications for the reduction to the linear case. In this section, we show that this desired simplification does indeed occur.

We first recall the state variable likelihood function (3.10), derived at the end of Section 3.2, which may be used to show that the joint density of the posterior distribution for the latent states, $\mathbf{x}$, and model parameters, $\boldsymbol{\theta}$, is given up to an unknown normalising constant, by

$$\begin{aligned} p(\mathbf{X}, \boldsymbol{\theta} \mid \boldsymbol{\phi}, \boldsymbol{\gamma}) &\propto p(\mathbf{X} \mid \boldsymbol{\theta}, \boldsymbol{\gamma}) p(\mathbf{X} \mid \boldsymbol{\phi}) \\ &\propto \prod_{k=1}^{K} \exp\left\{ -\frac{1}{2}\mathbf{x}_k^T \mathbf{C}_{\boldsymbol{\phi}_k}^{-1} \mathbf{x}_k - \frac{1}{2}(\mathbf{f}_k - \mathbf{m}_k)^T \mathbf{S}_{\boldsymbol{\phi}_k}^{-1} (\mathbf{f}_k - \mathbf{m}_k) \right\} p(\boldsymbol{\theta}). \end{aligned} \tag{3.17}$$

The ease with which we can sample from the posteriors corresponding to the joint density (3.17) is therefore directly related to the function $f(\mathbf{x}, \boldsymbol{\theta})$ governing the dynamics of the proposed ODE model. In fact on inspection we see that if the flow contains polynomial terms in the components x_{in}, $i = 1, \ldots, K$ and $n = 1, \ldots, N$ of the N-vector $\mathbf{x}_i$ of maximum degree P then the posterior conditional will consist of an exponential, with argument that is a degree $2P$ polynomial in the state variables. Density functions of this form for $P \geq 2$ have received relatively little attention aside from the univariate case when the exponential contains a degree four polynomial in a single variable. This is sometimes referred to as the exponential quartic model and has been considered by [Fisher, 1922] in the context of frequency curves of Pearsonian type distribution, more detail concerning the calculation of the normalising constant and moments is provided in [O'Toole, 1933] and the maximum likelihood estimator has been considered by [Matz, 1978] in the context of constructing bimodal distributions.

The extension to the multivariate case is significantly hampered by the difficulty in deriving the normalising constant. Some progress has been made by considering the case where the multivariate polynomials are homogenous allowing expressions for the normalising constant in terms of certain invariants. This approach is analogous to the role played by the determinant in the quadratic case, and some progress has been made in [Morozov and Shakirov, 2009], however, at this point exact solutions are available only for relatively modest values of the dimension N. While our interest in the remainder of this thesis will be in the linear case, it is worth mentioning that given these comments it seems unlikely, even in the case where polynomials give the nonlinearities of the flow function, that it will be possible to derive closed-form expressions for the posteriors corresponding to the density (3.17). The same holds if we wished to consider polynomial transforms of the model parameters — for instance, to ensure positivity or as Taylor series approximations to more general functions. Extensions of this kind would be a worthwhile area of future research since exact results, or accurate approximations would allow the MH sampling scheme at the end of the previous section to be replaced by a Gibbs update, even in the nonlinear polynomial setting.

Returning to the specific setting of the MLFM introduced in Section 2.3 our interest is in the case where the flow is linear in the state variables, that is a polynomial of degree one. It follows from the discussion above that the resulting posterior conditional will be an exponential quadratic, and so the state variables have a conditional Gaussian distribution given the remaining model parameters. This same reasoning applies to any additional parameters that enter the flow function in a linear manner so we may conclude that for any parameter, or state variable, entering the flow function linearly there will be a corresponding set of variables such that the act of conditioning upon this set will result in a conditional Gaussian distribution for this variable. This conclusion will also hold when the evolution equation is nonlinear in the state variables but linear in the model parameters in which case we will still get a Gaussian conditional for the parameters.

In this section we will provide further details of this claim for the MLFM introduced in Section 2.3 by adapting the method in the previous section to the case where the model parameter $\boldsymbol{\theta}$ represents the latent force variables and additional structural parameters of the model. The corresponding evolution equation is given by

$$\begin{aligned} f(\mathbf{x}(t), \boldsymbol{\theta}) &= f(\mathbf{x}(t), \mathbf{g}, \mathbf{B}) \\ &= \left(\mathbf{A}_0 + \sum_{r=1}^{R} \mathbf{A}_r g_r(t) \right) \mathbf{x}(t) \\ &= \left(\sum_{d=1}^{D} \beta_{0d} \mathbf{L}_d + \sum_{r=1}^{D} g_r(t) \sum_{d=1}^{D} \beta_{rd} \mathbf{L}_d \right) \mathbf{x}(t). \end{aligned} \tag{3.18}$$

We shall see that both the latent states and the latent forces possess conditionally Gaussian posterior distributions, and we shall derive the explicit formulas for the mean and covariance of these conditional densities. Since we can solve for the conditional densities exactly, it will be possible to replace the MH scheme introduced at the end of the previous section with the corresponding Gibbs sampling scheme using the specific conditionals. However, the presence of exact conditionals also suggests the possibility of replacing the Monte Carlo sampling method with a variational approximation, [Bishop, 2006], where the tractable Gaussian form of the conditional distributions will make it possible to apply a mean-field factorisation for the approximating density. We defer the details of this procedure until Chapter 5 where we shall also introduce an expectation maximisation (EM) algorithm for carrying out maximum likelihood or maximum a posteriori inference to the model introduced in this chapter.

3.3.1 Posterior conditional distributions

Our first step in deriving the parameters of the conditional distributions for the latent state and model parameters is to rewrite (3.18) with equivalent representations each of which displays an emphasis on the latent states, latent forcing functions and connection coefficients respectively depending on which of these variables we wish to derive the conditional distribution of. As in Section 3.2 the vector $\mathbf{f}_k$ is taken to be N-dimensional with entries $\mathbf{f}_{ki}$ given by the kth component of the $\mathbb{R}^K$ valued output $f(\mathbf{x}(t_i), \boldsymbol{\theta})$ of the evolution equation, for convenience we state the components of the vector-valued function (3.18) given above as

$$f_k(t) = \sum_{j=1}^{K} A_{0kj} x_j(t) + \sum_{r=1}^{R} g_r(t) \sum_{j=1}^{K} A_{rkj} x_j.$$

We also let $\mathbf{x}$ denote the full $(N \times K)$-vector of latent states given by vectorising the $N \times K$ array $X_{ni} = [\mathbf{x}(t_n)]_i$. We shall make use of the elementwise product of two vectors of the same size, $\mathbf{u}$ and $\mathbf{v}$, defined by $(\mathbf{u} \circ \mathbf{v})_i := \mathbf{u}_i \mathbf{v}_i$. Using this notation we are able to give three equivalent representations of the vectors $\mathbf{f}_k$, in doing so it will be convenient to also define the auxillary vector $\mathbf{g}_0$ by $\mathbf{g}_{0n} = 1$ for $n = 1, \ldots, T$. From the linearity of (3.18) we have

$$\begin{aligned}
\mathbf{f}_k &= \sum_{j=1}^{K} \sum_{r=0}^{R} A_{rkj} \mathbf{g}_r \circ \mathbf{x}_j \\
&= \sum_{j=1}^{K} \mathbf{u}_{kj} \circ \mathbf{x}_j && (3.19a) \\
&= \mathbf{v}_{k0} + \sum_{r=1}^{R} \mathbf{v}_{kr} \circ \mathbf{g}_r && (3.19b) \\
&= \sum_{r=0}^{R} \sum_{d=1}^{D} \beta_{rd} \mathbf{w}_{krd}, && (3.19c)
\end{aligned}$$

where A_{rkj} is the kj-th element of $\mathbf{A}_r = \sum_{d=1}^{D} \beta_{rd} \mathbf{L}_d$ and we have defined the NK-vectors

$$\begin{aligned}
\mathbf{u}_{kj} &= \sum_{r=0}^{R} A_{rkj} \mathbf{g}_r, && (3.20a) \\
\mathbf{v}_{kr} &= \sum_{j=1}^{K} A_{rkj} \mathbf{x}_j, && (3.20b) \\
\mathbf{w}_{krd} &= \mathbf{g}_r \circ \sum_{j=1}^{K} L_{dkj} \cdot \mathbf{x}_j, && (3.20c)
\end{aligned}$$

and L_{dkj} is the (k, j)-entry of the dth basis matrix $\mathbf{L}_d$.

Posterior conditionals for the latent state variables

We first use these representation (3.19a) to rewrite the argument of the exponential in (3.17) as an exponential quadratic in the state variables. By conditioning on the set of vectors $\{\mathbf{u}_{kj}\}$, and by extension the variables $\mathbf{g}_r, \mathbf{A}_r$ from which these vectors are composed, we observe that the argument of the exponential in (3.17) is a quadratic.

Explicitly we can rearrange the argument of the exponential in (3.17) to give

$$\begin{aligned}
-\frac{1}{2}\Big((\mathbf{f}_k-\mathbf{m}_k)^T\mathbf{S}_{\boldsymbol{\phi}_k}^{-1}(\mathbf{f}_k-\mathbf{m}_k)\Big) = -\frac{1}{2}\Bigg(&\sum_{i=1}^{K}\sum_{j=1}^{K}(\mathbf{u}_{ki}\circ\mathbf{x}_i)^T\mathbf{S}_{\boldsymbol{\phi}_k}^{-1}(\mathbf{u}_{kj}\circ\mathbf{x}_j \\
&-\sum_{i=1}^{K}(\mathbf{u}_{ki}\circ\mathbf{x}_i)^T\mathbf{S}_{\boldsymbol{\phi}_k}^{-1}\mathbf{M}_k\mathbf{x}_k \\
&-\sum_{i=1}^{K}\mathbf{x}_k^T\mathbf{M}_k^T\mathbf{S}_{\boldsymbol{\phi}_k}^{-1}(\mathbf{u}_{ki}\circ\mathbf{x}_i) \\
&+\sum_{i=1}^{K}\sum_{j=1}^{K}\mathbf{x}_i^T\mathbf{M}_k^T\mathbf{S}_{\boldsymbol{\phi}_k}^{-1}\mathbf{M}_k\mathbf{x}\Bigg), \qquad (3.21)
\end{aligned}$$

which is equivalent to the quadratic form

$$-\frac{1}{2}\Big((\mathbf{f}_k-\mathbf{m}_k)^T\mathbf{S}_{\boldsymbol{\phi}_k}^{-1}(\mathbf{f}_k-\mathbf{m}_k)\Big) = -\frac{1}{2}\mathbf{x}^T\boldsymbol{\Lambda}_k\mathbf{x}, \qquad (3.22)$$

where $\boldsymbol{\Lambda}_k := \boldsymbol{\Lambda}_k(\mathbf{g},\mathbf{B},\boldsymbol{\phi},\boldsymbol{\gamma})$ is the $NK\times NK$ block diagonal matrix with entries given by the matrices

$$\begin{aligned}
\boldsymbol{\Lambda}_{kij}(\mathbf{g},\mathbf{B},\boldsymbol{\phi},\boldsymbol{\gamma}) =&\mathbf{u}_{ki}\mathbf{u}_{kj}^T\circ\mathbf{S}_{\boldsymbol{\phi}_k}^{-1} \\
&-\delta_{ki}\operatorname{diag}(\mathbf{u}_{ki})\mathbf{S}_{\boldsymbol{\phi}_k}^{-1}\mathbf{M}_k-\delta_{kj}\mathbf{M}_k^T\mathbf{S}_{\boldsymbol{\phi}_k}^{-1}\operatorname{diag}(\mathbf{u}_{kj}) \\
&+\delta_{ki}\delta_{kj}\mathbf{M}_k^T\mathbf{S}_{\boldsymbol{\phi}_k}^{-1}\mathbf{M}_k, \qquad (3.23)
\end{aligned}$$

where for an N-vector $\mathbf{v}$ we define diag($\mathbf{v}$) to be the $N\times N$ diagonal matrix with the main diagonal given by $\mathbf{v}$ and zero elsewhere. Letting the arbitrary parameter $\boldsymbol{\theta}$ in (3.17) be replaced by the case where $\boldsymbol{\theta}$ is given by the $N\times R$ vector of latent forces, $\mathbf{g}$, and using 3.22 we may conclude that

$$p(\mathbf{x}\mid\mathbf{g},\mathbf{B},\boldsymbol{\phi},\boldsymbol{\gamma}) = \mathcal{N}\left(\mathbf{x}\mid\mathbf{0},\left(\boldsymbol{\Lambda}_{ode}+\mathbf{C}_{\boldsymbol{\phi}}^{-1}\right)^{-1}\right), \qquad (3.24)$$

where $\mathbf{C}_{\boldsymbol{\phi}}$ is the $NK\times NK$ block-diagonal matrix where the diagonal elements are the prior covariance matrices $\mathbf{C}_{\boldsymbol{\phi}_k}$ for each dimensional component which by assumption are independent. The matrix $\boldsymbol{\Lambda}_{ode} := \boldsymbol{\Lambda}_{ode}(\mathbf{g},\mathbf{B},\boldsymbol{\phi},\boldsymbol{\gamma})$ is the contribution to the likelihood from the ODE model and is given by summing over the terms given in (3.23), that is

$$\boldsymbol{\Lambda}_{ode}(\mathbf{g},\mathbf{B},\boldsymbol{\phi},\boldsymbol{\gamma}) = \sum_{k=1}^{K}\boldsymbol{\Lambda}_k(\mathbf{g},\boldsymbol{\phi}_k,\boldsymbol{\gamma}_k), \qquad (3.25)$$

and we shall also define the covariance matrix appearing in (3.24) by

$$\mathbf{K}_x(\mathbf{g},\mathbf{B},\boldsymbol{\phi},\boldsymbol{\Gamma}) = \Big(\boldsymbol{\Lambda}_{ode}(\mathbf{g},\mathbf{B},\boldsymbol{\phi},\boldsymbol{\Gamma})+\mathbf{C}_{\phi}^{-1}\Big)^{-1}. \qquad (3.26)$$

Therefore, before considering the data, we have shown that the distribution of the trajectory variables is given by a mean zero Gaussian distribution, the covariance matrix of which depends on the GP hyperparameters, $\boldsymbol{\phi}$, and the regularisation parameters $\boldsymbol{\gamma}$, as well as the latent force and connection coefficient variables.

To complete the derivation of the posterior distribution we also need to include the observed data points. The conditional Gaussian structure of the trajectory variables combined with the additive Gaussian error model for the observed variables, and using standard properties of the Gaussian distribution provided in Appendix A.2, we have

$$\begin{aligned} p(\mathbf{x} \mid \mathbf{y}, \boldsymbol{\tau},\ \mathbf{g}, \boldsymbol{\phi}, \boldsymbol{\gamma}, \mathbf{B}) &\propto p(\mathbf{Y} \mid \mathbf{x}, \boldsymbol{\tau}) p(\mathbf{x} \mid \mathbf{g}, \mathbf{B}, \boldsymbol{\phi}, \boldsymbol{\gamma}) \\ &= \mathcal{N}\Big(\mathbf{y} \mid \mathbf{x}, \mathrm{diag}(\boldsymbol{\sigma}) \otimes \mathbf{I}_T\Big) \mathcal{N}\left(\mathbf{x} \mid \mathbf{0}, \left(\boldsymbol{\Lambda}_{ode} + \mathbf{C}_{\boldsymbol{\phi}}^{-1}\right)^{-1}\right) \\ &= \mathcal{N}\left(\mathbf{x} \mid \boldsymbol{\Sigma}[\mathrm{diag}(\boldsymbol{\tau}) \otimes \mathbf{I}_T]\mathbf{y}, \boldsymbol{\Sigma}\right) \end{aligned} \tag{3.27}$$

where we have defined the vector of precision parameters $\boldsymbol{\tau}$ by $[\boldsymbol{\tau}]_k = \sigma_k^{-2}$, for $k = 1, \ldots, K$, and the covariance matrix $\boldsymbol{\Sigma} = \boldsymbol{\Sigma}(\mathbf{g}, \boldsymbol{\beta}, \boldsymbol{\gamma}, \boldsymbol{\phi}, \boldsymbol{\tau})$ is defined by

$$\boldsymbol{\Sigma}^{-1} = \boldsymbol{\Lambda}_{ode}(\mathbf{g}, \mathbf{B}, \boldsymbol{\gamma}, \boldsymbol{\phi}) + \mathbf{C}_{\boldsymbol{\phi}}^{-1} + \mathrm{diag}(\boldsymbol{\tau}) \otimes \mathbf{I}_T, \tag{3.28}$$

where $\boldsymbol{\Lambda}$ is given by (3.25).

Posterior condtional for the latent force variables

We can apply the same methods as used in the previous section to construct the conditional distribution for the latent forces conditional on the latent states and additional model parameters, only this time we use the representation given by (3.19b). Because of the presence of the offset matrix, $\mathbf{A}_0$, and the fact that the predicted mean $\mathbf{m}_k$ has no functional dependence on the latent forces means that, unlike the case for the latent states, we can not directly expand the argument of the exponential in (3.17) as a homogenous quadratic form. Instead we have

$$\mathbf{f}_k - \mathbf{m}_k = \sum_{r=1}^{R} \mathbf{v}_{kr} \circ \mathbf{g}_r - (\mathbf{m}_k - \mathbf{v}_{k0}),$$

and therefore

$$\begin{aligned} (\mathbf{f}_k - \mathbf{m}_k)^\top \mathbf{S}_k^{-1} (\mathbf{f}_k - \mathbf{m}_k) = &\sum_{r=1}^{R}\sum_{s=1}^{R} (\mathbf{v}_{kr} \circ \mathbf{g}_r)^\top \mathbf{S}_k^{-1} (\mathbf{v}_{kr} \circ \mathbf{g}_s) \\ &- \sum_{r=1}^{R} (\mathbf{v}_{kr} \circ \mathbf{g}_r)^\top \mathbf{S}_k^{-1} (\mathbf{m}_k - \mathbf{v}_{k0}) \\ &- \sum_{s=1}^{R} (\mathbf{m}_k - \mathbf{v}_{k0})^\top \mathbf{S}_k^{-1} (\mathbf{v}_{ks} \circ \mathbf{g}_s) \\ &+ (\mathbf{m}_k - \mathbf{v}_{k0})^\top \mathbf{S}_k^{-1} (\mathbf{m}_k - \mathbf{v}_{k0}). \end{aligned} \tag{3.29}$$

The final term does not depend on $\mathbf{g}$ and so we may ignore it in deriving the conditional density. Rearranging all those terms with an explicit dependance on the latent force variables we have the quadratic expression

$$\mathbf{g}^\top \mathbf{V}_k^\top \mathbf{S}_k^{-1} \mathbf{V}_k \mathbf{g} - 2\mathbf{g}^\top \mathbf{V}_k^\top \mathbf{S}_k^{-1} (\mathbf{m}_k - \mathbf{v}_{k0}) + \text{const.}, \tag{3.30}$$

where we have defined the $N \times NR$ block matrix $\mathbf{V}_k$ by

$$\mathbf{V}_k = \left[\operatorname{diag}(\mathbf{v}_{k1}) | \cdots | \operatorname{diag}(\mathbf{v}_{kR})\right].$$

To derive the conditional distribution we must also add the quadratic contribution from the prior, $\mathbf{g}^\top \mathbf{C}_{\boldsymbol{\psi}}^{-1}\mathbf{g}$, and from that we can conclude by identifying terms that the posterior conditional of the latent force variable will be a Gaussian with density

$$p(\mathbf{g} \mid \mathbf{X}, \mathbf{B}, \boldsymbol{\phi}, \boldsymbol{\psi}, \boldsymbol{\gamma}) = \mathcal{N}\left(\mathbf{g} \mid \mathbf{m}_g(\mathbf{X}, \mathbf{B}, \boldsymbol{\phi}), \mathbf{K}_g(\mathbf{X}, \mathbf{B}, \boldsymbol{\phi}, \boldsymbol{\psi})\right), \tag{3.31}$$

where the mean and variance parameters dependent on the latent trajectory variables as well as the hyperparameters of the latent force and are given explicitly by

$$\mathbf{m}_g(\mathbf{X}, \mathbf{B}, \boldsymbol{\phi}, \boldsymbol{\psi}, \boldsymbol{\gamma}) = \mathbf{K}_g(\mathbf{X}, \mathbf{B}, \boldsymbol{\phi}, \boldsymbol{\psi})^{-1} \sum_{k=1}^{K} \mathbf{V}_k^\top \mathbf{S}_k^{-1}(\mathbf{m}_k - \mathbf{v}_{k0}) \tag{3.32}$$

$$\mathbf{K}_g(\mathbf{X}, \mathbf{B}, \boldsymbol{\phi}, \boldsymbol{\psi}, \boldsymbol{\gamma}) = \left(\sum_{k=1}^{K} \mathbf{V}_k^\top \mathbf{S}_k^{-1}\mathbf{V}_k + \mathbf{C}_{\boldsymbol{\psi}}^{-1}\right)^{-1}. \tag{3.33}$$

This allows us to perform accurate conditional sampling of the latent forces, but it is clear that the dependence of the moments of this distribution on the state variables and the hyperparameters will be nonlinear. Therefore it will not be possible to perform direct marginalisation over either the state or hyperparameter terms and we instead we must consider alternative methods, such as MCMC sampling if we wish to construct the marginal posteriors $p(\mathbf{g} \mid \mathbf{Y})$. We also note the appearance of the hyperparameter of the latent trajectory variables, $\boldsymbol{\phi}$, in the posterior conditional distribution of the latent force variable and the regularisation term $\boldsymbol{\gamma}$. Gauging the dependence of the distribution on these parameters is challenging, and we shall discuss some of the implications of this feature later on in this chapter.

Posterior conditional for β_{rd}

We now make use of the representation (3.19c), and then similar to the process above we can take the log-transform of the joint density to obtain the quadratic

$$\begin{aligned}
-\frac{1}{2}\sum_{k=1}^{K}(\mathbf{f}_k - \mathbf{m}_k)^\top \mathbf{S}_k^{-1}(\mathbf{f}_k - \mathbf{m}_k) = -\frac{1}{2}\bigg(&\boldsymbol{\beta}^\top \sum_{k=1}^{K} \mathbf{W}_k^\top \mathbf{S}^{-1}\mathbf{W}_k\boldsymbol{\beta} \\
&- 2\boldsymbol{\beta}^\top \sum_{k=1}^{K} \mathbf{W}_k^\top \mathbf{S}_k^{-1}\mathbf{m}_k\bigg) \\
&+ \text{const.},
\end{aligned} \tag{3.34}$$

where $\mathbf{W}_k$ is the $(R+1)D \times N$ matrix where the $(rD+d)$th row with indices $r = 0, 1, \ldots, R$ and $d = 1, \ldots, D$ corresponds to the vector $\mathbf{w}_{krd}$. Up to this point we have not explicitly specified a prior for the connection coefficients, if we specify a Gaussian prior

$$p(\mathbf{B} \mid \boldsymbol{\zeta}) = \mathcal{N}(\boldsymbol{\beta} \mid \mathbf{0}, \mathbf{C}_{\zeta}), \tag{3.35}$$

for some vector of hyperparameters $\boldsymbol{\zeta}$. Then after identifying coefficients in (3.34) then we may conclude, see Section B.2 in the appendix, that $\boldsymbol{\beta}$ has a Gaussian distribution with mean denoted by $\mathbf{m}_\beta$ and covariance matrix denoted by $\mathbf{C}_\beta$. Compactly we write

$$p(\mathbf{B} \mid \mathbf{X}, \mathbf{g}, \boldsymbol{\phi}, \boldsymbol{\gamma}, \boldsymbol{\zeta}) = \mathcal{N}\left(\boldsymbol{\beta} \mid \mathbf{m}_\beta, \mathbf{K}_\beta\right), \tag{3.36}$$

with the parameters given explictly by

$$\mathbf{m}_\beta(\mathbf{X}, \mathbf{g}, \boldsymbol{\phi}, \boldsymbol{\gamma}, \boldsymbol{\zeta}) = \mathbf{K}_\beta \sum_{k=1}^{K} \mathbf{W}_k^\top \mathbf{S}_k^{-1} \mathbf{m}_k, \tag{3.37a}$$

$$\mathbf{K}_\beta(\mathbf{X}, \mathbf{g}, \boldsymbol{\phi}, \boldsymbol{\gamma}, \boldsymbol{\zeta}) = \left(\sum_{k=1}^{K} \mathbf{W}_k^\top \mathbf{S}_k^{-1} \mathbf{W}_k + \mathbf{C}_\zeta^{-1} \right)^{-1}. \tag{3.37b}$$

Posterior conditional distribution for the observation noise $\boldsymbol{\sigma}$.

We recall from Section 3.2.1 that the distribution of the observations is assumed to be given by Gaussian independent error models with a different standard derivation σ_k for each of the $k = 1, \ldots, K$ dimensional components, although a more general Gaussian additive error models would not significantly complicate matters. In practice it will be easier to work with the vector of precisions $\boldsymbol{\tau}$ with $\tau_k = \sigma_k^{-2}$. If we give each τ_k an independent Gamma prior

$$\tau_k \sim \text{Gamma}(a_{k0}, b_{k0}), \qquad k = 1, \ldots, K \tag{3.38}$$

then conditional on the collection of the observed data $\mathbf{y}$ and the latent states $\mathbf{x}$ we have the conditional independence properties

$$p(\boldsymbol{\tau} \mid \mathbf{Y}, \mathbf{X}, \mathbf{g}, \mathbf{B}) = p(\boldsymbol{\tau} \mid \mathbf{Y}, \mathbf{X}) = \prod_{k=1}^{K} p(\tau_k \mid \mathbf{y}_k, \mathbf{x}_k),$$

since the observation noise model is centred on the latent state variables, and separated from the latent force variables after conditioning on the states. Using the observation noise model (3.2) we see that the chosen prior is the usual conjugate prior, and that the conditional posterior distribution is given by

$$\begin{aligned} p(\tau_k \mid \mathbf{y}_k, \mathbf{x}_k) &= \prod_{i=1}^{T} \mathcal{N}(y_k(t_i) \mid x_k(t_i), \tau_k^{-1}) \\ &\propto \tau_k^{N/2 + a_{k0} - 1} \exp\Big(-\tau_k (b_{k0} + (\mathbf{x}_k - \mathbf{y}_k)^\top (\mathbf{x}_k - \mathbf{y}_k)) \Big), \end{aligned} \tag{3.39}$$

which is a Gamma distribution with parameters

$$a_k = a_{k0} + T/2 \tag{3.40}$$

$$b_k = b_{k0} + (\mathbf{x}_k - \mathbf{y}_k)^\top (\mathbf{x}_k - \mathbf{y}_k), \tag{3.41}$$

and this distribution is straightforward to sample from.

Posterior conditional distributions for the latent state hyperparameters, $\boldsymbol{\phi}, \boldsymbol{\gamma}$

While the variables considered up to this point possessed tractable conditional posteriors this property will not in general be true for the hyperparameters of the latent state Gaussian processes interpolators. Instead it is typically only possible to derive the unnormalised density

$$p(\boldsymbol{\phi}, \boldsymbol{\gamma} \mid \mathbf{Y}, \mathbf{X}, \mathbf{g}, \boldsymbol{\phi}, \boldsymbol{\gamma}) \propto p(\mathbf{X} \mid \mathbf{g}, \boldsymbol{\phi}, \boldsymbol{\gamma}) p(\boldsymbol{\phi}) p(\boldsymbol{\gamma}), \tag{3.42}$$

where the conditional Gaussian density $p(\mathbf{X} \mid \mathbf{g}, \boldsymbol{\phi}, \boldsymbol{\gamma})$ is given by the ODE response model conditional Gaussian (3.24), the mean and covariance of which depend on $\boldsymbol{\gamma}, \boldsymbol{\phi}$ through the inverse of the covariance matrices $\mathbf{S}_k$ and the conditional mean functions $\mathbf{m}_{\dot{x}_k \mid x_k}$. This is a complex, nonlinear transformation, and so does not lead to an analytically tractable conditional distribution for these variables. Instead sampling methods would need to be used to sample these variables from the posterior (3.42), for instance using MH methods as done in [Dondelinger et al., 2013] Unfortunately each evaluation of the probability density function is expensive requiring the K inversions of the prior covariance matrices $\mathbf{C}_{\phi_k}$, and the K inversions of the covariance matrices $\mathbf{S}_{\phi_k, \gamma_k}$.

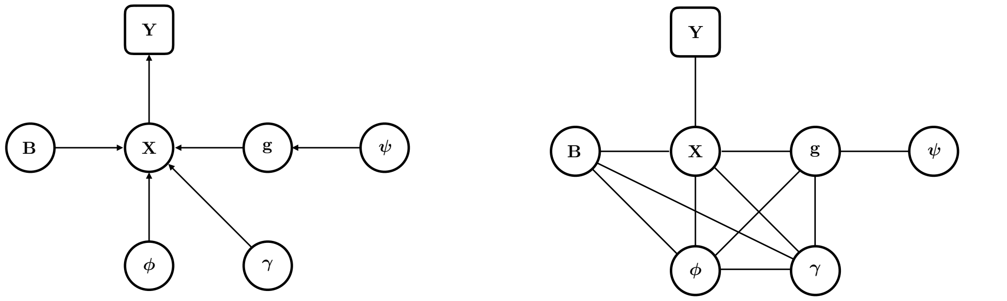
The analytical intractability arises after marginalisation over the latent gradient variables, indeed we shall see below that the hyperparameters of the latent force GPs have relatively simple posteriors in comparison. The marginalisation over the gradients causes the latent state GP hyperparameters to become densely connected in the graph corresponding to the conditional independence properties of the adaptive gradient matching model, and this connectivity is represented in Figure 3.2 as well as being apparent from the inspection of the density (3.17).

As discussed at the end of the previous section this expensive update motivated the choice in [Calderhead et al., 2009] to update the Gaussian process hyperparameter terms using the Gaussian process prior model, rather than the full model, in effect this would allow us to ignore the terms arising from the gradient process $\mathbf{S}_{\phi_k}$ and $\mathbf{M}_{\phi_k}$ when updating these variables in some sampling scheme. In standard GP regression, the hyperparameters play a relatively unimportant role, in that the conditional distributions are relatively robust to minor changes in these variables [Rasmussen and Williams, 2006]. However for more complex models, and in particular for this model, where the role of the GP as an interpolant has an essential role in maintaining structural information the inference of these variables becomes more important, and we are less inclined to use this approximation. The problematic nature of these hyperparameters does complicate inference, not only for sampling methods but also for the deterministic variational methods we consider in Chapter 5. We shall return to this subject frequently throughout this work, not only in the context of the adaptive gradient matching method but also discussing the similar problems in the method we introduce in the next chapter.

Posterior conditional distributions for the latent force hyperparameters, $\boldsymbol{\psi}$

In contrast to the hyperparameters for the latent state Gaussian process interpolators which were densely connected in the resulting conditional independence network, the hyperparameters of the latent Gaussian processes enjoy better conditional independence properties as is clear from their separations in Figure 3.2a. Therefore the posterior conditional density factors as

$$\begin{aligned} p(\psi \mid \mathbf{y}, \mathbf{x}, \mathbf{g}, \boldsymbol{\gamma}, \boldsymbol{\psi}) &= p(\psi \mid \mathbf{g}) \\ &\propto p(\mathbf{g} \mid \psi) p(\psi) \\ &\propto \prod_{r=1}^{R} p(\mathbf{g}_r \mid \psi_r) p(\psi_r), \end{aligned} \tag{3.43}$$



(a) Pre-moralisation network graph.

(b) Post-moralisation network graph.

Figure 3.2: Graphical representation of the decomposition of the joint distribution $p(\mathbf{y}, \mathbf{X}, \mathbf{B}, \mathbf{g}, \boldsymbol{\phi}, \boldsymbol{\gamma}) = p(\mathbf{y} \mid \mathbf{X})p(\mathbf{X} \mid \mathbf{B}, \mathbf{g}, \boldsymbol{\phi}, \gamma)p(\mathbf{B})p(\mathbf{g})p(\boldsymbol{\phi})p(\boldsymbol{\gamma})$ of the variables in the adaptive gradient matching method. After moralisation the conditional distribution of the state interpolant hyperparameter $\boldsymbol{\phi}$ and regularisation parameter $\boldsymbol{\gamma}$ become densely connected with the variables $\mathbf{g}$ and $\mathbf{B}$ of principal interest.

The conditional density (3.43) is exactly the conditional density function of R independent Gaussian process regression models. Therefore we can apply the usual methods, [Williams and Rasmussen, 1996], for hyperparameter inference in Gaussian process regression models for sampling from this distribution. Although in general this density is not available in closed form, because of the nonlinear dependence of the kernel function on the hyperparameters, it is a standard problem and as such has existing implementations in statistical software.

3.4 Marginalisation over the latent states

In proposing the MLFM our emphasis has been on learning the matrix valued process $\mathbf{A}(t)$ which specifies the dynamics, and this is done by inferring the distribution of the pair $\{\mathbf{g}, \mathbf{B}\}$. Since the latent states are not of immediate interest during the model training stage it is worth addressing the possibility of achieving a more efficient learning method by first marginalising over these variables. In this section we show that this marginalisation is possible, and discuss whether this step is beneficial for the resulting inferential process.

Since our interest is in marginalising over the latent states in order to infer the latent forces and connection parameters we will simplify our notation by suppressing the dependence of the conditional distributions on the set of parameters $\{\boldsymbol{\gamma}, \boldsymbol{\phi}, \boldsymbol{\psi}, \boldsymbol{\zeta}\}$, so that in what follows we have $p(\cdot) := p(\cdot \mid \boldsymbol{\gamma}, \boldsymbol{\phi}, \boldsymbol{\psi}, \boldsymbol{\zeta})$.

Combining the observation noise model (3.2) described in Section 3.2.1 with the conditional density of the latent state variables (3.27) we perform the marginalisation

$$\begin{aligned} p(\mathbf{y} \mid \mathbf{g}, \mathbf{B}, \boldsymbol{\sigma}) &= \int p(\mathbf{y}, \mathbf{X} \mid \mathbf{g}, \mathbf{B}, \boldsymbol{\sigma}) \mathrm{d}\mathbf{X} \\ &= \int p(\mathbf{y} \mid \mathbf{X}, \boldsymbol{\sigma}) p(\mathbf{y} \mid \mathbf{g}, \mathbf{B}) \mathrm{d}\mathbf{X} \\ &= \int \mathcal{N}(\mathbf{y} \mid \mathbf{x}, \operatorname{diag}(\boldsymbol{\sigma}^{\circ 2}) \otimes \mathbf{I}_N) \mathcal{N}(\mathbf{x} \mid \mathbf{0}, \mathbf{K}_x(\mathbf{g}, \mathbf{B})) \mathrm{d}\mathbf{x}, \end{aligned} \tag{3.44}$$

where we have defined $\boldsymbol{\sigma}^{\circ 2}$ to be the elementwise squaring operation, similarly we define

$\boldsymbol{\tau}^{\circ -1}$ to be the elementwise inverse of a vector. This integral is tractable and leads to

$$p(\mathbf{y} \mid \mathbf{g}, \mathbf{B}, \boldsymbol{\sigma}) = \mathcal{N}(\mathbf{y} \mid \mathbf{0}, \operatorname{diag}(\boldsymbol{\sigma}^{\circ 2}) \otimes \mathbf{I}_N + \mathbf{K}_x(\mathbf{g}, \mathbf{B})). \tag{3.45}$$

While this marginalisation step was straight forward we would now like to use Bayes rule and then obtain MAP estimates, or samples of, the variables with posterior conditional given by

$$p(\mathbf{g}, \mathbf{B} \mid \mathbf{y}, \boldsymbol{\sigma}) \propto p(\mathbf{y} \mid \mathbf{g}, \mathbf{B}, \boldsymbol{\sigma}) p(\mathbf{B}) p(\mathbf{g}). \tag{3.46}$$

Recalling from (3.26) that the entries of the inverse of the matrix $\mathbf{K}_x(\mathbf{g}, \mathbf{B})$ appearing in (3.45), are quadratic in the parameters $\mathbf{g}$ and $\mathbf{B}$ we can quickly conclude the conditional distribution $p(\mathbf{g}, \mathbf{B} \mid \mathbf{y})$ is likely in to be intractable. To sample from this distribution it would therefore be necessary to consider approximation methods, including Metropolis-Hasting methods as used in the general setting of the adaptive gradient matching methods as discussed earlier in this chapter, or gradient based sampling methods such as Langevin diffusion sampling [Besag, 1994] or hybrid Monte-Carlo methods [Duane et al., 1987]. The loss of the exact conditionals in more complex setting is likely to offset any efficiency gained by avoiding sampling of the latent states. This is certainly true in terms of numerical efficiency for each step of the chosen sampler although we do acknowledge that there is the potential for methods which update the complete set of parameters at each step to exhibit better mixing than the conditional Gibbs samplers in some applications.

If however we are only interested in constructing point estimates such as the maximum a posteriori (MAP) estimates of the model parameters then avoiding the optimisation of the NK latent state variables will be more efficient. In this instance constructing the gradients of (3.45) with respect to the parameters $\mathbf{g}$ and $\mathbf{B}$ will require computing the gradient of $\boldsymbol{\Lambda}_{ode}$, which would also be necessary in an optimisation of the complete density

$$p(\mathbf{X}, \mathbf{g}, \mathbf{B} \mid \mathbf{y}) \propto p(\mathbf{y} \mid \mathbf{X}, \boldsymbol{\sigma}) p(\mathbf{X} \mid \mathbf{g}, \mathbf{B}), \tag{3.47}$$

so that at least as much work is being done in optimising the complete model leading to efficiency gains for marginalised density (3.46). Although we do note that since

$$\frac{\partial \log p(\mathbf{X}, \mathbf{g}, \mathbf{B} \mid \mathbf{y}, \boldsymbol{\sigma})}{\partial \mathbf{x}} = -(\operatorname{diag}(\boldsymbol{\tau}) \otimes \mathbf{I}_T)(\mathbf{y} - \mathbf{x}) - \mathbf{K}_x(\mathbf{g}, \mathbf{B})^{-1}\mathbf{x}, \tag{3.48}$$

the actual decrease in computation complexity of a particular gradient step using the marginal density is relatively modest so long as the dimension K remains relatively small which is true for the examples we consider in this thesis. Instead the computational bottleneck will be in calculating the $(R+1) \cdot D + T \cdot R$ gradients of $\mathbf{K}_x$ with respect to the variables $\mathbf{B}$ and $\mathbf{g}$ respectively. With relatively little additional work needing to be done to compute the gradient (3.48) once $\mathbf{K}_x$ has been evaluated. Aside from the computational efficiency the more parsimonious parameterisation of (3.46) is also likely to lead to a better conditioned optimisation.

In summary for constructing point estimates of the model parameters it is more efficient to first marginalise over the state variables. This step is only possible because of the linear ODE setting and is not available in the more general setting. If however we wish to sample from the posterior then we find it preferable to sample from the complete model, and so better exploit the conditional independence structure discussed in Section

3.3.1. These observations are similar to those we shall experience in Chapter 5 when we introduce an EM method for constructing the MAP estimate for the adaptive gradient matching method, which provides no benefit over working with the marginalised density just discussed, but that when constructing a distributional estimate using a variational method it is useful to work with the full model, and so benefit from the conditional independence structure.

3.5 Discussion

In this chapter, we have presented our first method for carrying out approximate inference in the latent force model with multiplicative interactions introduced in Chapter 2. We have made use of the Bayesian adaptive gradient matching methods developed in [Calderhead et al., 2009, Dondelinger et al., 2013], which in turn built on the work of [Varah, 1982, Ramsay et al., 2007] for carrying out approximate likelihood inference for ODE models. In the remainder of this thesis we shall refer to the MLFM with approximate density given by the adaptive gradient matching approximation as the MLFM-AG method.

In the general setting of nonlinear ODEs with finite dimensional parameters, this class of methods motivates an approximate joint density up to an unknown normalising constant as given by (3.17). In this chapter, we have demonstrated that for the case of linear ODEs with infinite dimensional parameters this joint density possesses tractable conditional posteriors. In particular for the state variables, the latent forces, and the connection coefficients it is possible to construct the Gaussian conditional distributions presented in Section 3.3.1. This simplification over the general setting allows Metropolis-Hastings updates to be replaced with Gibbs sampling methods. Not only does the conditional independence structure lend itself well to a Gibbs sampling routine for these variables, but also naturally suggests a mean field variational sampling method which we will describe in Chapter 5 after introducing our second approximation to the model.

The approximation was made possible using the produce of experts approximation in [Calderhead et al., 2009] and represented graphically in Figure 3.1. A crucial ingredient is the GP prior over the state variable, which embeds smoothness properties and further specifies a prior on the gradient of the process. It is simple to see that even considering the simplest example of the MLFM that there is a certain amount of cognitive dissonance in the specification. The simplest example of the univariate MLFM is sufficient to ensure that in general the support of the MLFM will not be Gaussian. One could then argue that the GP prior is assigned before any knowledge of the ODE model, but at no point is there any belief update that encodes these hard constraints.

The validity, or othrwise, of the GP prior expert is not merely a philosophical point but is instead connected with an important potential shortcoming of this class of approximations. The structural information is embedded into the model by convolution with the gradient expert. As such while the gradient expert matches the model in the tangent space, the GP interpolator, and in particular its hyperparameters, retains an important role in determining this tangent space. When the data is densely observed this tangent space is well constrained, and so the GP expert has a reduced role and correspondingly the sensitivity of the model to different values of the hyperparameter $\boldsymbol{\phi}$ is much reduced. However, when data is sparse relative to the model complexity there are a great many different possible interpolating functions, each of which will imply a different tangent space. As discussed in 3.2.2 the original specification of the

Bayesian gradient matching procedure as introduced by [Calderhead et al., 2009] contained no mechanism by which the hyperparameters of the interpolating process, and so by extension the interpolating process itself, could be influenced by the dynamical system structure. The extension to the adaptive gradient matching model of [Dondelinger et al., 2013] made a significant step in correcting this defect, however the overall success of the method is still strongly linked with the accuracy of the interpolant, and even with the gradient of this interpolant now being better regularised by the model parameters the method is still limited to a first order local correction. As a consequence there remains the strong possibility of an increasing deterioration in the performance of the MLFM-AG method when the method is applied to datasets obtained at lower sampling frequencies, and indeed we observe this directly in Section 6.2.4 when we investigate the performance of this approximation using simulation studies, and we provide some further remarks on this property in Section 8.2.1.

Ultimately, the class of gradient matching approaches do not lead to proper generative models, a phenomena we have already encountered when commenting on the dissonance between the specification of a GP prior and the strong structure preservation of certain dynamical systems. In an attempt to correct for this [Barber and Wang, 2014] introduced an alternative model which they demonstrated could be specified by a proper generative belief network, and they referred to their method as the GPODE model. However, the generative structure of the GPODE model is achieved only after the imposition of certain independence assumptions which, while not immediately obvious in the initial presentation, was made clearer in the presentation by [Macdonald et al., 2015], in particular they stressed that the simple structure of this model as been gained only at the expense of a false equivalence between the elimination of a variable inside a probabilistic model, and a marginalisation over that variable. Consequently, the state trajectories were entering the probabilistic model in two distinct locations, and therefore any satisfactory identification of these variables would destroy the appealing chain structure of this model. As a direct impact they note that this can lead to identifiability problems when data were systematically missing, while the gradient matching approaches do not suffer from this issue, and as such we do not consider the GPODE model further in this thesis. While it is philosophically disquieting that the product of experts assumption (3.9), which underlies the gradient matching paradigm, cannot be formulated in terms of a proper probabilistic generative model, this methodological limitation is well offset by the significant computational advantages provided by this class of models – particularly in the case of infinite dimensional latent parameters we are considering.

Given the comments above regarding the probabilistic structure of the MLFM-AG method our work in the next chapter may be viewed as a more principled attempt to construct a distribution of the latent force that respects the generative structure, i.e. that the latent force is a random variable formed from a certain transformation of a random initial condition and a set of random latent forces. While we must also turn to approximations and not exact solutions we will manage to avoid some of the difficulties of hyperparameter inference, but that additional regularisation and tuning parameters will need to be included.

Despite these caveats, the adaptive gradient matching approximation leads to an inference scheme which, while being more complex than the GP regression model for the LFM, is nevertheless tractable. The replacement of the exactly Gaussian structure by the greater complexity of the conditional Gaussian structure of the MLFM-AG approximation is well compensated for by the greater control of the model geometry. We observe from our simulation studies that these methods can give accurate results, particularly to first order moments, when the sampling frequency is relatively high — as we might expect due to the reduced possibility for misspecification of the interpolating state variable as the distance between points decreases. It is tempting to conjecture that this conditional Gaussian structure is the best case scenario as we move past the LFM structure, and we shall see that the method introduced in the next chapter shares this feature.

Chapter 4

Approximate solution using the Neumann series expansion

4.1 Introduction

Our work in the previous chapter introduced an approximation to the MLFM which was able to avoid the need to ever explicitly solve the underlying ODE. This allowed us to derive a method for carrying out inference in which the GP framework of the LFM was replaced with a tractable conditional Gaussian structure. In order to circumvent the problem of solving the ODE the MLFM-AG method relied on the introduction of GP interpolants of the trajectory, and its gradient, which were then compared with the parameteric evolution equation. As a consequence of the use of interpolants we remarked that it would be reasonable to suppose that the performance of these methods will be influenced by the frequency with which we have collected observations, and we shall see in Chapter 6 that this is the case.

In this chapter we introduce an alternative method that more closely follows the construction of the distribution of the LFM in Section 2.2.1 by expressing the trajectory as a transformation of the driving stochastic variables. The relevant transformation in the LFM was a simple linear integral equation, unfortunately such a simple transformation is no longer possible once we allow multiplicative interactions and instead we must consider more complex transformations, such as the series expansion discussed in Section 2.3.1. By truncating the series expansion after a finite number of terms we introduce an adjustable order parameter intended to allow for more accurate solutions over longer time intervals with relatively sparse data – precisely the scenario in which we have justifiable concerns about the appropriateness of the GP interpolant approximation underlying the MLFM-AG model.

To help motivate the method in this chapter we note that an important component in the development of the adaptive gradient matching methods is the use of the gradient expert to impose an approximate fixed point condition of the linear differential operator given by

$$\mathcal{D}f(t) \triangleq \frac{\mathrm{d}f(t)}{\mathrm{d}t} - \mathbf{A}(t)f(t),$$

where $\mathbf{A}(t) = \mathbf{A}(g_1(t), \ldots, g_r(t))$ is a sample path realisation of the matrix-valued Gaussian processes specifying our MLFM, and where each of the component functions will be smooth. In Section 3.2 this fixed point condition was enforced on the conditional distribution through the introduction of the gradient expert with the temperature parameters, $\boldsymbol{\gamma}$, governing the extent to which the triple $\{\boldsymbol{\theta}, \mathbf{X}, \dot{\mathbf{X}}\}$ was allowed to deviate from an exact solution to this linear operator equation at the observed time points. In this chapter we consider an analogous construction, but rather than consider the differential operator we proceed by first integrating the ODE and so consider the alternative linear transformation given by the integral operator

$$\mathcal{L}f(t) \triangleq f(t_0) + \int_{t_0}^{t} \mathbf{A}(\tau)f(\tau)\mathrm{d}\tau. \tag{4.1}$$

That an appropriate solution of the latent state trajectory should satisfy this integral equation will act as our motivation for introducing a novel alternative method to that considered in the previous chapter for producing approximate solutions to the MLFM.

Our approximation will attempt to concentrate the conditional distribution of the trajectory around a fixed point of the operator (4.1). Similar to the use of the gradient expert in the previous section we will be applying the relevant operator conditional on the coefficient matrix, and therefore treating the operator as a given deterministic operator. The choice to consider conditional fixed points the method of this, and the previous, chapter avoid the problem of discussing the existence of marginal solutions of random integral, or equivalently differential, equations, which in general is a hard problem as we discussed in Section 2.3.1. The downside of course is the relative inefficiency of conditional inference compared to inference after marginalisation over the latent forces as is possible for the LFM.

When applied to the trajectories of the MLFM the state variable appears on both sides of (4.1) defining the solution only implicity and so is not as immediately useful as the explicit fixed point of the gradient expert in the previous chapter. In Section 4.2 we discuss how a solution to this integral equation can be constructed by an iterative method leading to the series expansion first discussed in Section 2.3.1. While this does achieve our initial goal of deriving an explicit expression for the trajectory variables as a transformation of the latent variables this series expansion is still impractical for the purposes of calculating posterior conditional distributions.

To rectify this we demonstrate in Section 4.3 the idea in [Tait and Worton, 2018] to retain a truncated set of successive approximations to the expansion of the trajectory as a “complete data” model, leading to a tractable approximation realised by restricting the priors to the intersection of certain linear subspaces. This will allow us to demonstrate Gaussian approximations to the conditional distributions of the variables of interest in our model similar to that which was possible for the adaptive gradient matching method. Following the development in the previous chapter we shall consider in Section 4.4 the possibility of marginalising over the approximation to the trajectory we have introduced, this step is particularly important because the model completion process leads to a significant increase in the parameterisation of the problem. Unfortunately we note that the marginal distribution has a relatively complex form and we defer the construction of efficient variational methods for handling this density until the next chapter. The methods developed in this chapter provide local approximations to the solution, and so for longer time intervals will require an increasingly higher order of approximation to the trajectory with a corresponding increase in the numerical complexity of the method. We attempt to reduce this scaling problem in Section 4.5 by replacing the process of modelling the trajectory over the whole interval using a single higher order method with an approximation in terms of a mixture of several lower order approximations. We conclude this chapter with a discussion and compare this method with that developed in the previous chapter.

4.2 The Neumann series expansion

In order to derive a series expansion for the sample paths of the MLFM model we are going to first present a brief discussion on the general solution to time dependent linear ODEs in the deterministic setting. Our interest is in the broadly defined class of initial value problems (IVP) described by

$$\frac{\mathrm{d}\mathbf{x}(t)}{\mathrm{d}t} = A(t)\mathbf{x}(t), \qquad t \geq 0, \qquad \mathbf{x}(0) = \mathbf{x}_0, \tag{4.2}$$

where $A(t)$ is some linear operator, and $\mathbf{x}_0$ is some variable in the domain of $A(t)$. For the MLFM we will be interested in the case where A is a $K \times K$ real valued matrix operator such that component functions $A_{ij}(t)$ are either constant or else linear combinations of smooth functions supported on the RKHS corresponding to each of the Gaussian processes latent forces. In this context we will denote the matrix operator by the bold font $\mathbf{A}(t)$, but the same formal solution will hold for much more general linear operators. Upon integrating (4.2) we may replace the differential equation with an equivalent integral equation which will be satisfied by a solution, $\mathbf{x}(t)$, to the IVP

$$\mathbf{x}(t) = \mathbf{x}(0) + \int_0^t A(\tau)\mathbf{x}(\tau)\mathrm{d}\tau. \tag{4.3}$$

The integral equation (4.3) integrates out the unobserved gradient process, but replaces the explicit linear relationship between the state variables and any parameters of the operator $A(t)$ with an implicit identity for the state variable. As such we have lost the conditionally linear link between the parameters and the gradient that were the fundamental reason for the analytic tractability of the adaptive gradient matching methods considered in the previous chapter. Therefore, while integration of the IVP removes the nuisance gradient term there are two immediate complications in (4.3); the presence of the integral and the fact that the state variable is given only up to an implicit relationship. By replacing the integral with a numerical quadrature the former issue can be handled in a reasonably straightforward way and therefore it is arguably the latter problem which presents the most serious conceptual barrier.

It is worth emphasising that our interest is not in the problem of solving (4.3) on the basis of a given realisation of the operator $A(t)$, but in constructing efficient methods for inverting a, potentially noisy, discrete realisation of the state variables in such a way that it is possible to construct an approximation to the conditional distribution of the component functions of the matrix operator. In the limit considering a continuous realisation of the process with no noise then the problem becomes degenerate and so the point estimate solving the integral equation and the, degenerate, conditional distribution become equivalent. From our perspective we must balance the existence of the solution at the degenerate limit with the reality of finitely observed data, the necessity of completing the estimator of the trajectory and the computational demands of this refinement.

While we have discounted the possibility of propagating a given random variable through a numerical solution operator as a reasonable approach to constructing a posterior conditional distribution it is worth remarking that the situation would be different if we were able to obtain a marginal distribution for any finite dimensional distribution of the process $\mathbf{x}(t)$. If we partition any such sample as $\mathbf{X} = \{\mathbf{X}_o, \mathbf{X}_c\}$ where $\mathbf{X}_o$ corresponds to time indices at which we expect to have a set of observations, and which we further assume to temporally sparse. The completion, $\mathbf{X}_c$, is assumed to be generated on a refinement of the interval which is dense enough to allow for accurate numerical quadrature.

Given a dense sample of the process then after replacing the integral in (4.3) with a quadrature we will realise a large system of linear equations which the trajectory satisfies, or equivalently a system of linear constraints on the the coefficient matrix. If we let $\mathbf{a}$ denote a vectorisation of the coefficient matrix, evaluated along the same partition, then this implies the existence of a system of linear equations

$$\mathbf{M}[\mathbf{X}]\mathbf{a} = \mathbf{b}[\mathbf{X}], \tag{4.4}$$

for some matrix $\mathbf{M}$ and vector $\mathbf{b}$ depending on the complete sample and quadrature weights. If we knew the conditional distribution

$$p(\mathbf{X}_c \mid \mathbf{X}_o) = \int p(\mathbf{X}_c, \mathbf{a} \mid \mathbf{X}_o)\mathrm{d}\mathbf{a}$$

then, conditional on the observations $\mathbf{X}_o$, we could consider the following heuristic description of a generative model for samples from the coefficient matrix

$$\mathbf{X}_c \sim P(\mathbf{X}_c \mid \mathbf{X}_o), \tag{4.5a}$$
$$\mathbf{a} \mid \mathbf{X}_c, \mathbf{X}_o \sim \delta(\mathbf{a} - (\mathbf{M}[\mathbf{X}]\mathbf{a} - \mathbf{b}[\mathbf{X}])), \tag{4.5b}$$

where samples from the degenerate distribution (4.5b) correspond to solving the system of linear equations. The important conceptual difference of this scheme when compared with the forward generative methods discussed in Section 1.1.5 is the ability to carry out the simulation in the data space, rather than a simulation of the coefficient matrix which is then propagated forward.

Unfortunately, we do not have access to this marginalised conditional distribution, and therefore no way to form the dense samples of the process. It is interesting to note that an approach with the same emphasis on first completing the paths in the data space, and then inferring parameters has been successfully developed to allow exact sampling in SDEs in [Beskos et al., 2006].

If we cannot recover the unknown marginal distribution we might still consider replacing the unknown distribution with a Gaussian approximation, for example the Gaussian distribution which matches the marginal moments obtained using the methods referenced in Chapter 2. However, these methods require a finite correlation time and given our assumption of sparsely observed data this assumption is unlikely to be satisfied. Even if it were possible to realise this idea concretely we would still need to solve the numerically involved equations for the moments and so we do not pursue this approach further. Nevertheless the motivating idea of realising the distribution of the components of the operator as the solution to least squares problems will be an important part of our construction in this chapter.

Because of the issues discussed above rather than attempting to solve (4.3) directly we consider an iterative method of solution which is a standard element of the classical existence and uniqueness theorems for ODEs, see for instance [Arnold, 1973]. The method is often referred to as Picard iteration, or the method of successive approximations, and involves iterating an mth order approximation, $\mathbf{x}^{(m)}(t)$, to the solution of (4.2) using the integral transform

$$\mathbf{x}^{(m)}(t) \mapsto \mathbf{x}^{(m+1)} \triangleq \mathbf{x}^{(m)}(t_0) + \int_{t_0}^{t} A(\tau)\mathbf{x}^{(m)}(\tau)\mathrm{d}\tau. \tag{4.6}$$

This method is applied starting from an initial approximation to the solution, $\mathbf{x}^{(0)}(t)$, which should be chosen to satisfy the IVP, i.e. $\mathbf{x}^{(0)}(t_0) = \mathbf{x}_0$. It is typical to take the initial approximation to be the constant valued function satisfying the initial value. We follow the presentation in [Iserles, 2004] and define the linear integral operator $\mathcal{K}$ acting on $\mathbb{R}^K$ valued functions, f, by

$$\mathcal{K}[f](t) \triangleq \int_0^t A(\tau)f(\tau)\mathrm{d}\tau, \tag{4.7}$$

then (4.3) may be rewritten as

$$(I - \mathcal{K})\mathbf{x}(t) = \mathbf{x}_0, \tag{4.8}$$

where I is the identity operator $If = f$. A formal solution to (4.8) may be obtained, [Atkinson, 1997], by inverting the operator $(I - \mathcal{K})$ by way of the series expansion

$$\mathbf{x}(t) = (I - \mathcal{K})^{-1}Y_0 = \sum_{m=0}^{\infty} \mathcal{K}^m \mathbf{x}_0, \tag{4.9}$$

where we have defined K^0 to be the identity transform and K^m is defined by repeated applications of (4.7), that is

$$\mathcal{K}^0\mathbf{x} = \mathbf{x}(t), \qquad \mathcal{K}^{m+1}\mathbf{x} = \int_0^t A(\tau)\mathcal{K}^m\mathbf{x}(\tau)\mathrm{d}\tau, \qquad m \geq 0, \tag{4.10}$$

so that the Picard iterates (4.6) are equivalent to $\mathbf{x}^{(n+1)} = \mathbf{x}_0 + \mathcal{K}\mathbf{x}^{(n)}$. For given operator $A(t)$ the formal expansion (4.9) converges provided $\|\mathcal{K}\| < 1$, which can always be guarantee by choosing t sufficiently small since the operator (4.7) is easily seen to be $\mathcal{O}(t)$. We briefly remark that by preconditioning the ODE (4.2) it may be possible to increase the order of this approximation, for example [Iserles, 2004] use preconditioning to achieve an $\mathcal{O}(t^2)$ approximation, but it is not immediately clear how to adapt this observation into the probabilistic setting in a way that does not lead to the resulting inference problem becoming intractable.

If we expand (4.9) and then collect terms we can derive an expansion of the solution of the form

$$\begin{aligned}\mathbf{x}(t) = \Bigg(I + &\int_{t_0}^t A(\tau)\mathrm{d}\tau \\ &+ \int_{t_0}^t \int_{t_0}^{\tau_1} A(\tau_1)A(\tau_2)\mathrm{d}\tau_1\mathrm{d}\tau_2 + \cdots \Bigg)\mathbf{x}_0,\end{aligned} \tag{4.11}$$

which we refer to as a Neumann series expansion, but is also known in matrix analysis as a Peano series expansion [Gantmacher, 1959], and in the physics literature as a Freeman-Dyson series [Dyson, 1949]. This is exactly the expansion introduced in Section 2.3.1 for motivating the approximation of [van Kampen, 1974] to the marginal moments of the MLFM.

The series expansion (4.11) gives our first explicit representation of the state variable as some transformation of the latent Gaussian process force variables. However in practice this is of limited use because of the complexity of the resulting expansion. Inspecting each of the nested integrals that constitute the terms of this expansion we see that the integrands are degree m polynomials in the latent force variables, these will therefore have complex product distributions to which we must then apply the linear integral transformations – constructing marginal moments would be challenging even with the aid of computer algebra systems, and the prospect of constructing a complete joint distribution for the state variables and the latent forces in this manner is daunting.

More useful is the appreciation that, for a known complete realisation of the operator $A(t)$, each successive iteration of (4.6) is a linear integral transformation of a current approximation, $\mathbf{x}^{(m)}(t)$, to a new approximation, $\mathbf{x}^{(m+1)}(t)$. This also implies that for known complete trajectories, $\mathbf{x}^{(m)}$ and $\mathbf{x}^{(m+1)}$, we can recover the matrix function $A(t)$ by applying linear operators. Combining these observations allows us to begin to realise the linear system of equations method discussed in the introduction to this chapter and in the next section we expand upon this approach.

4.2.1 The Magnus series expansion

We have focused on the Neumann series expansion (4.9), or equivalently (4.11), because it constructs a solution directly in the ambient data space $\mathbb{R}^K$, an alternative approach is the Magnus series expansion, [Magnus, 1954], which provides a series expansion of a matrix valued function $\Omega(t)$ such that

$$\mathbf{x}(t) = e^{\Omega(t)}\mathbf{x}_0,$$

is a solution to the IVP (4.2), that is

$$\frac{\mathrm{d}}{\mathrm{d}t}e^{\Omega(t)}\mathbf{x}(t) = A(t)e^{\Omega(t)}\mathbf{x}_0.$$

It can be shown that an appropriate expansion of $\Omega(t)$ may be formed by integral transforms of $A(t)$ and its commutators and therefore resides in the same Lie algebra as $A(t)$. Since this element of the Lie algebra is then mapped in to the state space through the exponential mapping the trajectories are necessarily members of the associated Lie group and so the Magnus series method preserves the model geometry exactly. In comparison the Neumann series expansion does not have this property and instead only preserves the geometry in the limit. The two expansions are related and it can be shown, [Aparicio et al., 2005], that the Magnus series is the formal logarithm of the Neumann series expansion.

The exact geometry preservation combined with the possibility of constructing approximations in a vector space would seem to make the Magnus series an appealing choice for constructing our approximations. However this would require working with the nonlinear transformation between the latent space and the data space given by the matrix exponential. In general the matrix exponential does not have a unique inverse, nor can we guarantee the pre-image will be countable [Culver, 1966]. Therefore constructing an approximation based around the Magnus series expansion would involve approximating a nested integral expansion, which will be at least as hard as the method we are considering for the Neumann series, and then dealing with nonlinear matrix exponential transformation. By constructing our approximation directly in the ambient data space we avoid the need to consider this transformation, and so sacrifice exact geometry preservation for tractable inference.

4.3 Method of successive approximations

4.3.1 Model specification

Having introduced the underlying series approximation which we shall use to construct our method of approximate inference we now present the model specification similar to the presentation given in Section 3.2.1. Many of the variables that we shall introduce will have roles analogous to those in the adaptive gradient matching, and the choice of variable names will reflect the similar roles.

We shall be interested in carrying out inference for the evolution of a K-dimensional latent state variable $\mathbf{x}(t)$ on the basis of a collection of observations $\mathbf{Y} = (\mathbf{y}(t_1), \ldots, \mathbf{y}(t_T))$ observed at a set of T time points with $t_1 < \ldots < t_T$. We shall take the observation distribution conditional on the latent state variables to be the same as that considered in the previous chapter. Restating here for convenience we assume that each $\mathbf{y}(t_i)$ is a noisy independent observation from a Gaussian distribution centred on the corresponding latent variable $\mathbf{x}(t_i)$ leading to the conditional distribution

$$\begin{aligned} p(\mathbf{Y} \mid \mathbf{X}, \boldsymbol{\sigma}) &= \prod_{i=1}^{T} p_{obs}(\mathbf{y}(t_i) \mid \mathbf{x}(t_i), \boldsymbol{\sigma}) \\ &= \prod_{i=1}^{T} \prod_{k=1}^{K} \mathcal{N}(\mathbf{y}_k(t_i) \mid x_k(t_i), \sigma_k^2). \end{aligned} \tag{4.12}$$

4.3.2 Discretisation of the integral operator

While we use the same choice of observation model as for the MLFM-AG method the approximation to the distribution of the latent states is significantly different. As previously we shall assume that $\mathbf{Y}$ is a noisy observation of the latent states, but that the latent state variable has been obtained by at least one iteration of the Picard transform (4.6). Therefore our proposal is to augment the set of model parameters with an additional set of variables which are to be interpreted as the preimages of the latent state variable under the successive applications of the Picard map.

Our first step is to replace the integral operator (4.7) with a numerical quadrature rule using some suitable discretisation. In order to carry out the process of quadrature accurately it may be necessary to augment the time set of observed variable times $\mathcal{T} = \{t_1, \ldots, t_T\}$ to an augmented set $\bar{\mathcal{T}} = \{\tau_1, \ldots, \tau_N\}$. The augmented set $\bar{\mathcal{T}}$ will be a finer partition of the interval $[t_1, t_T]$ so that $t_1 = \tau_1 < \ldots < \tau_N = t_T$ where $T < N$, and we further assume that for each time t_i for which we have an observed data point there is a corresponding variable in the augmented set. We now apply a numerical quadrature rule to replace the integral operator (4.7) with the quadrature

$$\int_{\tau_{i-1}}^{\tau_i} A(\tau)\mathbf{x}(\tau)\mathrm{d}\tau \approx (\tau_i - \tau_{i-1}) \left(\theta A(\tau_{i-1})\mathbf{x}(\tau_{i-1}) + (1-\theta)\right) A(\tau_i)\mathbf{x}(\tau_i), \tag{4.13}$$

for $i = 2, \ldots, N$ and for some parameter $\theta \in [0, 1]$, [Baker, 2000]. In practice we typically choose the value $\theta = 1/2$ which coincides with the trapezoidal rule. It would be tempting to consider higher order quadrature rules, and so minimise the extent to which we need to augment the variable set; however, the construction of higher order quadrature formulas is in general not straightforward, and in this work, we only consider the simple trapezoidal approximation.

For a linear ODE such as the MLFM, we can provide an explicit representation of the operator so as make clear the dependence on random variables $\mathbf{g}$ and $\mathbf{B}$, and to efficiently provide gradients of the model with respect to these parameters. In this chapter we shall assume that the vector form of the state variable is arranged as $\mathbf{x} = (\mathbf{x}(\tau_1), \ldots, \mathbf{x}(\tau_N))^\top$, which differs from the vectorisation in the previous chapter in that we now vectorise along the rows of $\mathbf{X}$.

We have assumed that there is some point which will correspond to the initial time for the IVP defining the MLFM, although we note that there is no requirement that this point corresponds to earliest point for which we have an observation. We denote the arbitrary initial time by t_ν where ν is some index in $\{1, \ldots, N\}$. Then applying the quadrature (4.13) to each subinterval of our refined partition we approximate the operator (4.7) by the quadrature

$$\begin{aligned} \mathcal{K}_\nu \mathbf{x}(\tau_n) &= \int_{t_\nu}^{t_n} \mathbf{A}(\tau)\mathbf{x}(\tau)\mathrm{d}\tau \\ &\approx \sum_{i=0}^{n} \mathbf{A}(\tau_i)\mathbf{x}(\tau_i) w_{in}, \end{aligned} \tag{4.14}$$

for some appropriate set of weights $\{w_{in}\}$. We can append additional zeros for all w_{ij} with $n+1 \leq j \leq N$, and then collect these variables in the $N \times N$ matrix, $\mathbf{W}$, which will depend on the choice of quadrature, the choice of initial time point and the partition of the time set. Therefore the quadrature approximation (4.14) may be rewritten as the matrix/vector product

$$\mathcal{K}_\nu \mathbf{x}(\tau_n) \approx \left[\mathbf{K}_\nu \mathbf{x}\right]_n,$$

where $[\mathbf{K}_\nu \mathbf{x}]_n$ is the nth subvector of the NK block vector. We also necessarily have that the weight $w_{\nu\nu} = 0$. Alternatively we may write

$$\begin{aligned} \mathbf{K}_\nu[\mathbf{g}, \boldsymbol{\beta}] &= \sum_{n=0}^{N} \mathbf{w}_n \mathbf{e}_n^\top \otimes \mathbf{A}(t_n) \\ &= \sum_{n=0}^{N} \mathbf{w}_n \mathbf{e}_n \otimes \left(\sum_{r=0}^{R} g_{rn} \mathbf{A}_r \right) \\ &= \sum_{n=0}^{N} \mathbf{w}_n \mathbf{e}_n \otimes \mathbf{A}_0 + \sum_{r=1}^{R} \sum_{n=0}^{N} g_{rn} \cdot \mathbf{w}_n \mathbf{e}_n \otimes \mathbf{A}_r \\ &= \sum_{n=0}^{N} \left(\sum_{d=1}^{D} \left[\beta_{0d} + \sum_{r=1}^{R} g_{rn}\beta_{rd} \right] \mathbf{w}_n \mathbf{e}_n^\top \otimes \mathbf{L}_d \right). \end{aligned} \tag{4.15}$$

While in practice we rarely would have been to form the full $NK \times NK$ matrix $\mathbf{K}$, this explicit expression makes it straightforward to realise the gradients

$$\frac{\partial}{\partial g_{rn}} \mathbf{K}_\nu[\mathbf{g}, \boldsymbol{\beta}] = \mathbf{w}_n \mathbf{e}_n^\top \otimes \sum_{d=1}^{D} \beta_{rd} \mathbf{L}_d, \tag{4.16}$$

$$\frac{\partial}{\partial \beta_{rd}} \mathbf{K}_\nu[\mathbf{g}, \boldsymbol{\beta}] = \left(\sum_{n=1}^{N} g_{rn} \mathbf{w}_n \mathbf{e}_n^\top \right) \otimes \mathbf{L}_d. \tag{4.17}$$

It follows that after vectorisation, and by linearity, we have the representations in terms of the Jacobians

$$\begin{aligned} \operatorname{vec}(\mathbf{K}) &= \tilde{\mathbf{v}}_0[\boldsymbol{\beta}] + \mathbf{V}[\boldsymbol{\beta}]\mathbf{g} && (4.18a) \\ &= \mathbf{W}[\mathbf{g}]\boldsymbol{\beta}, && (4.18b) \end{aligned}$$

where $\tilde{\mathbf{v}}_0[\boldsymbol{\beta}] = \operatorname{vec}(\mathbf{K}_\nu[\mathbf{0},\boldsymbol{\beta}]$, and $\mathbf{V}$ is an $(NK)^2 \times NR)$ matrix where the columns are given by vectorising the gradient (4.16). Similarly $\mathbf{W}$ is the $(NK)^2 \times (R+1)D$ matrix with columns given by the vectorisation of the gradient (4.17).

We can easily extend our discrete representation of the operator $\mathcal{K}_\nu$ to a discrete version of the Picard map (4.6). For an index $\nu \in \{1,\dots,N\}$ we have the representation

$$\mathbf{x}(\tau_\nu) + \int_{\tau_\nu}^{t_n} \mathbf{A}(\tau)\mathbf{x}(\tau)\mathrm{d}\tau \approx [\mathbf{P}_\nu \mathbf{x}]_n, \qquad (4.19)$$

where we have defined the matrix operator $\mathbf{P}_\nu$ given by

$$\mathbf{P}_\nu \mathbf{v} = \mathbf{v}_\nu \otimes \mathbf{1}_N + \mathbf{K}_\nu[\mathbf{g},\boldsymbol{\beta}]\mathbf{v}, \qquad (4.20)$$

which is defined to act on block-vectors of the form $\mathbf{v} = (\mathbf{v}_1,\dots,\mathbf{v}_N)^\top$, with each subvector $\mathbf{v}_i \in \mathbb{R}^K$, for $i = 1,\dots,N$, such that $[\mathbf{K}_\nu[\mathbf{g},\mathbf{B}]\mathbf{v}]_\nu = \mathbf{v}_\nu$. By construction this operator preserves the property of the Picard iteration of leaving the initial condition invariant. Of course the representations (4.18) can be used to give affine representations of $\operatorname{vec}(\mathbf{P}_\nu)$ in terms of the latent force or connection coefficient variables, in particular we have

$$\begin{aligned} \operatorname{vec}(\mathbf{P}_\nu) &= \operatorname{vec}(\mathbf{E}_{\nu\nu}^{(N)} \otimes \mathbf{I}_K) + \operatorname{vec}(\mathbf{K}) \\ &= \mathbf{v}_0[\boldsymbol{\beta}] + \mathbf{V}[\boldsymbol{\beta}]\mathbf{g} && (4.21a) \\ &= \mathbf{w}_0 + \mathbf{W}[\mathbf{g}]\boldsymbol{\beta}, && (4.21b) \end{aligned}$$

where $\mathbf{E}_{ii}^{(N)}$ is the $N \times N$ matrix with entries $[\mathbf{E}_{ii}^{(N)}]_{nm} = \delta_{in}\delta_{im}$ and we have defined

$$\begin{aligned} \mathbf{v}_0 &= \tilde{\mathbf{v}}_0[\boldsymbol{\beta}] + \operatorname{vec}(\mathbf{E}_{\nu\nu}^{(N)} \otimes \mathbf{I}_K) \\ \mathbf{w}_0 &= \operatorname{vec}(\mathbf{E}_{\nu\nu}^{(N)} \otimes \mathbf{I}_K). \end{aligned}$$

The construction of these representations depends implicitly upon the fixed initial time, in what follows we shall suppress this dependence for notational convenience, apart from in Section 4.5 when we consider mixtures of different initial starting times.

The linear representations we have just introduced will act in a way analogous to those derived in Chapter 3 for the adaptive gradient matching approach. In particular, they will allow us to reconsider the motivating linear system of equations (4.4), but now between successive iterates of the Picard map. Anticipating the development in the next section consider a set of vectors $\mathbf{z}^{(m)}$, defined by the successive application of the Picard map

$$\mathbf{z}^{(m)} = \mathbf{P}[\mathbf{g},\boldsymbol{\beta}]\mathbf{z}^{(m-1)}, \qquad m = 1,\dots,M,$$

which after vectorisation, and using the representations (4.18), are equivalent to the system of equations

$$(\mathbf{z}^{(m-1)} \otimes \mathbf{I})\mathbf{V}[\boldsymbol{\beta}]\mathbf{g} = (\mathbf{z}^{(m-1)} \otimes \mathbf{I})[\tilde{\mathbf{v}}_0 + \mathbf{z}_\nu \otimes \mathbf{1}_N] - \mathbf{z}^{(m)}, \qquad (4.22)$$

and

$$(\mathbf{z}^{(m-1)} \otimes \mathbf{I})\mathbf{W}[\mathbf{g}]\boldsymbol{\beta} = (\mathbf{z}^{(m-1)} \otimes \mathbf{I})[\mathbf{z}_\nu \otimes \mathbf{1}_N] - \mathbf{z}^{(m)}. \tag{4.23}$$

By conditioning on the set of latent variables, and then rearranging these expressions we can construct systems of linear equations which the variables $\mathbf{g}$ and $\mathbf{B}$ should satisfy. Because in practice we do not have access to this full set of successive observations, but instead only a noisy observation of the limiting trajectory $\lim_{m\rightarrow\infty} \mathbf{z}^{(m)}$ we will necessarily have to augment our variable set if we are going to utilise this motivating idea of exploiting the conditional linear system of equations structure. We view this as a missing data problem and the full set of successive approximations as the complete data for our model. In attempting to introduce these variables we will require their distribution, and so run into the same difficulty as discussed previously when trying to introduce the distribution of the trajectory. However by making certain assumptions regarding the conditional distribution of each of the successive approximates conditional on those previous it will be easier to motivate a distribution for the complete set, and so overcome the barriers in the heuristic scheme (4.5) and we present the details of this approach in the next section.

4.3.3 Conditional successive approximations

To use the linear representations obtained by discretising the integral transforms we will need to augment our model with a collection of variables representing the successive approximations, in practice we will also need to truncate the series expansion (4.9) after a finite number of terms. We will see that successfully doing this will allow us to consider the constraining system of linear equations discussed in the previous section, but to do so we need to first construct a suitable probabilistic model for these variables that will allow them to be incorporated into the inference procedure. This process is analogous to that considered in the previous chapter, where it was necessary to augment the dataset with a representation of the gradient process.

We therefore introduce the collection of latent variables $\{\mathbf{Z}_0, \mathbf{Z}_1 \ldots, \mathbf{Z}_M\}$ with $\mathbf{Z}_m = (\mathbf{z}(\tau_1), \ldots, \mathbf{z}(\tau_T))$, for $m = 0, 1, \ldots, M$ and $\mathbf{Z}_M$ is identified with $\mathbf{X}$. We shall assume that any finite sample of the $\mathbf{z}^{(0)}$ of the process $\mathbf{Z}^{(0)}(t)$ is almost surely Gaussian distributed, and the sample paths are continuous with respect to the temporal argument, this includes GPs with appropriately defined kernel functions, but it also allows for the specification of regression type models for the distribution including constant initial approximations. We shall also typically assume independence across dimensions for the initial condition, as was done for the GP prior on the state variable in the previous chapter, that is we take the vectors $\mathbf{z}_k^{(0)} = (\mathbf{z}_k^{(0)}(\tau_1), \ldots, \mathbf{z}_k^{(0)}(\tau_N))^\top$, to be independent for $k = 1, \ldots, K$. In general we may write the distribution of the initial approximation $\mathbf{Z}^{(0)} = \{\mathbf{z}_1^{(0)}, \ldots, \mathbf{z}_K^{(0)}\}$ as

$$\begin{aligned} p(\mathbf{z}^{(0)} \mid \boldsymbol{\phi}) &= \prod_{k=1}^{K} p(\mathbf{z}_k^{(0)}) \\ &= \prod_{k=1}^{K} \mathcal{N}\left(\mathbf{z}_k^{(0)} \mid \mathbf{0}, \mathbf{C}_{\phi_k}\right), \end{aligned} \tag{4.24}$$

where each $\mathbf{C}_{\phi_k}$ is a postive-semi definite matrix, allowing for the possibility of the degenerate case with $\operatorname{rank}(\mathbf{C}_{\phi_k}) < N$. As with the introduction of the GP prior in the adaptive gradient matching methods discussed in the previous chapter, the choice of covariance function will determine the smoothness properties of the initial approximation to the state. Where this method will differ is that because the initial approximation is further removed from the data, this will not translate as directly into smoothness assumptions on the observed trajectory itself. It is because of these similarities with the role of the GP interpolators in the MLFM-AG method that we use the same symbol, ϕ_k, to parameterise the hyperparameters of the initial state interpolants.

There is much flexibility in the specification of this term, but the higher the order of the expansion the less it will depend on the initial value. With this in mind rather than specify a flexible, but hard to identify Gaussian processes, it may be simpler to consider finite basis regression models; indeed one choice is to fix the initial trajectory to be the constant model

$$\mathbf{z}^{(0)}(\tau_n) = \boldsymbol{\mu}_0, \qquad \forall n = 1, \ldots, N, \tag{4.25}$$

where $\boldsymbol{\mu}_0$ is an $\mathbb{R}^K$ valued random variable with distribution

$$p(\boldsymbol{\mu}_0) = \prod_{k=1}^{K} \mathcal{N}(\mu_{0k} \mid 0, \phi_k^{-1}), \tag{4.26}$$

and ϕ_k is a positive scalar. This allows us to specify the distribution of the constant initial value approximation of the variable, $\mathbf{z}^{(0)}$, as the product of the degenerate distributions

$$p(\mathbf{z}_k^{(0)} \mid \phi_k, \alpha_k) = \mathcal{N}(\mathbf{z}_k^{(0)} \mid \mathbf{0}, \phi_k^{-1}\mathbf{1}_N\mathbf{1}_N^\top), \tag{4.27}$$

where $\mathbf{1}_N$ is the N-vector with each entry equal to unity. This choice mirrors the deterministic setting of the Neumann series expansion discussed in the previous section where the initial approximation was taken to be a constant valued function agreeing with the initial condition specifying the IVP. In practice when updating the hyperparameters $\boldsymbol{\phi}$ we will work with the scalar random variables $[\mathbf{z}_k^{(0)}]_\nu$ corresponding to the fixed point indices, rather than the full N-vector.

A linear dynamic system model for the successive approximations

Having specified the initial distribution, then the interpretation of the collection of variables $\{\mathbf{Z}^{(m)}\}_{m=0}^{M}$ should be clear; they are each intended to represent an mth order approximation to the state variable $\mathbf{X}$ conditional on an observation of the latent force parameters and additional structural parameters. By analogy with the Picard map 4.6 we assume these variables are related by the linear mapping

$$\mathbf{z}^{(m)} = \mathbf{P}[\mathbf{g}, \mathbf{B}]\mathbf{z}^{(m-1)} + \mathbf{w}_m, \qquad m = 1, \ldots, M, \tag{4.28}$$

where $\mathbf{w}_m$ is some additional correction term, including the possibility of taking $\mathbf{w}_m = \mathbf{0}$ almost surely. A natural interpretation of this correction term is as the error introduced by replacing the random integral operator $\mathcal{K}$ with the discrete operator $\mathbf{K}$.

This interpretation is challenging; however, because as a random variable, we will have multiple sources of error. These errors include the error structure as a random variable, a deterministic error corresponding to the use of quadrature, and furthermore, these errors will be correlated with one another. However, since for suitably dense realisations of the trajectory, the quadrature error disappears we can construct a suitable approximation by using the independent error model and concluding that the majority of the informational content is contained in the linear map. (4.28) is captured by applying the matrix operator (4.20), and therefore in the case of dense realisations of the trajectory where the error is small, there will be minimal loss of information if we replace the correlated error term with an independent additive Gaussian noise term leading to the proposed transition distribution

$$\mathbf{z}^{(i+1)} \mid \mathbf{z}^{(i)}, \mathbf{g}, \mathbf{B}, \mathbf{\Gamma} \sim \mathcal{N}\left(\mathbf{z}^{(i+1)} \mid \mathbf{P}[\mathbf{g}, \mathbf{B}]\mathbf{z}^{(i)}, \mathbf{\Gamma}\right). \tag{4.29}$$

A similar use of quadrature is proposed in [Lawrence et al., 2006] applied to the integral operator (2.7) of the LFM considered in Section 2.2, however with a nonlinear transformation of the Gaussian process in the integrand. In that approach, no effort is made to proxy for the quadrature error, and so it effectively gets absorbed into the GP term.

This option is also open to us, corresponding to the choice of $\mathbf{\Gamma} = 0$. However, we shall see in the next section that $\mathbf{\Gamma}$ acts as a regularising term for the conditional distribution of the latent force variables and connection coefficients. As such it may be preferable to retain it in which case it plays a role analogous to the parameter $\boldsymbol{\gamma}$ introduced in the adaptive gradient matching method.

In this thesis this is the emphasis we stress; the transition density is to be viewed as a regularised approximation of a deterministic transformation. This specification acts to counterbalance the rigidity imposed on the model by working near the degenerate limit of almost complete sample paths, much as the introduction of the regularisation parameters in the previous chapter to counterbalance the hard constraint implied by the time-evolution equation. For this reason we denote this noise term by $\mathbf{\Gamma}$, and in application we shall typically assume this takes the simpler form

$$\mathbf{\Gamma} = \operatorname{diag}(\boldsymbol{\gamma}) \otimes \mathbf{I}_N,$$

rather than a fully specified covariance matrix.

Joint distribution of the successive approximations

Under the additive independent error assumption (4.29) we can treat the collection of successive approximations as a Markov chain in the order parameter leading to a joint conditional distribution given by

$$\begin{aligned} p(\mathbf{z}^{(0)}, \ldots, \mathbf{z}^{(m)} \mid \mathbf{g}, \mathbf{B}, \mathbf{\Gamma}) &= p(\mathbf{z}^{(0)}) \prod_{m=1}^{M} p(\mathbf{z}^{(m)} \mid \mathbf{z}^{(m-1)}, \mathbf{g}, \mathbf{B}, \mathbf{\Gamma}) \\ &= \mathcal{N}(\mathbf{z}^{(0)} \mid \mathbf{0}, \mathbf{C}_{\phi}) \prod_{m=1}^{M} \mathcal{N}(\mathbf{z}^{(m)} \mid \mathbf{P}[\mathbf{g}, \mathbf{B}]\mathbf{z}^{(m-1)}, \mathbf{\Gamma}), \end{aligned} \tag{4.30}$$

To complete the model specification we now need to link the data and the approximations to the latent states. This must be done both for the initial state approximation and for the final state. The latter is achieved by directly identifying the highest order approximation $\mathbf{z}^{(M)}$ with the trajectory variable directly leading to the emission density given by the observation noise distribution discussed above

$$\begin{aligned} p(\mathbf{Y} \mid \mathbf{Z}, \boldsymbol{\sigma}) &= p(\mathbf{Y} \mid \mathbf{Z}^{(M)}, \mathbf{g}, \mathbf{B}) \\ &= \prod_{k=1}^{K} \prod_{n=1}^{N} \mathcal{N}(\mathbf{y}_{kn} \mid \mathbf{z}_{kn}^{(M)}, \sigma_k^2). \end{aligned}$$

We remark that it is necessary to view the observation at the fixed initial time point in a manner distinct from the other trajectory variables. The distribution of state variables at the initial time will not depend on the parameters of $\mathbf{P}$ because of the fixed initial condition property. To elaborate on this point if $\nu \in \{1, \ldots, N\}$ is the index of the initial time then from the transition model (4.29) we have that $\mathbf{z}_{\nu}^{(m)} = \mathbf{z}_{\nu}^{(m-1)} + \mathbf{w}_{m\nu}$ so that the distribution of $\mathbf{z}_{\nu}^{(m)}$ has no direct dependence on the variables $\mathbf{G}$ and $\mathbf{b}$, and this is true for all $m = 1, \ldots, M$. It follows from this property, and if the covariance matrix $\boldsymbol{\Gamma}$ has small diagonal entries, that the distribution of $\mathbf{Z}^{(0)}(t_{\nu})$ will be well informed by the observations $\mathbf{Y}(t_{\nu})$ and so we specify the joint distribution of the observation at the fixed initial time by

$$p(\mathbf{Y}(\tau_{\nu}), \mathbf{Z}^{(0)}(\tau_{\nu}) \mid \mathbf{g}, \mathbf{B}, \boldsymbol{\tau}, \boldsymbol{\mu}_0, \boldsymbol{\phi}) = p(\mathbf{Y}(\tau_{\nu}) \mid \mathbf{z}^{(0)}(\tau_{\nu})), \tag{4.31}$$

where the distribution of $\mathbf{Z}^{(0)}$ is given by either (4.24) or (4.27). The conditional independence structure of this model is represented in Figure 4.2. In which we can clearly observe the linear Gaussian dynamic model arising from the additive error approximation. The state variables will have a Markov structure, and in the conditional independence graph, Figure 4.1, will be connected to the data through an emission distribution connecting the final state and the full set of observations, and an emission distribution relating the approximation of the initial state and the observation $\mathbf{Y}(\tau_{\nu})$. If we take the constant initial state approximation then we obtain the emission distribution

$$\begin{aligned} p(\mathbf{Y} \mid \mathbf{Z}^{(0)}, \boldsymbol{\sigma}) &= \prod_{k=1}^{K} p(\mathbf{Y}^{(0)}(\tau_{\nu}) \mid \mathbf{Z}(\tau_{\nu})^{(0)}, \boldsymbol{\sigma}) \\ &= \prod_{k=1}^{K} \mathcal{N}(Y_k(\tau_{\nu}) \mid Z_k^{(0)}(\tau_{\nu}), \sigma_k^2), \end{aligned} \tag{4.32}$$

where $\mathbf{Z}^{(0)}(\tau_{\nu})$ has a distribution given by (4.27).

If we then combine these two emission densities with the density of the state variables then we have the complete specification

$$p(\mathbf{Y}, \mathbf{Z} \mid \mathbf{g}, \mathbf{B}, \boldsymbol{\tau}, \boldsymbol{\Gamma}) = p(\mathbf{Y}(\tau_{\nu}) \mid \mathbf{Z}^{(0)}, \boldsymbol{\tau}) p(\mathbf{Y} \mid \mathbf{Z}^{(M)} \mid \mathbf{g}, \mathbf{B}, \boldsymbol{\Gamma}) p(\mathbf{Z} \mid \mathbf{g}, \mathbf{B}, \boldsymbol{\Gamma}), \tag{4.33}$$

the conditional Independence structure of this model is displayed in Figure 4.2, and takes the form a linear Gaussian dynamical system with emission densities connecting data to the first and final states.

Posterior conditional of the latent force variables

Having introduced an approximate density for the set of successive approximations to the state variable we now follow the development of the previous chapter to construct the posterior conditionals for the model parameters.

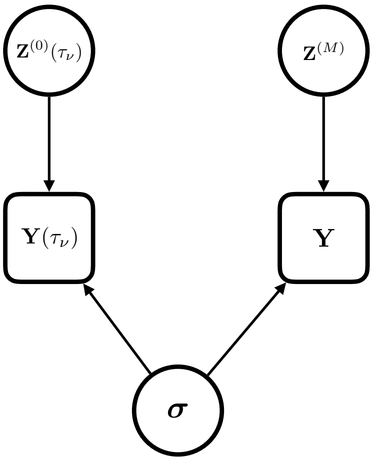


Figure 4.1: Conditional indepdence structure of the emission distributions in the linear dynamical system structure of the MLFM-SA model

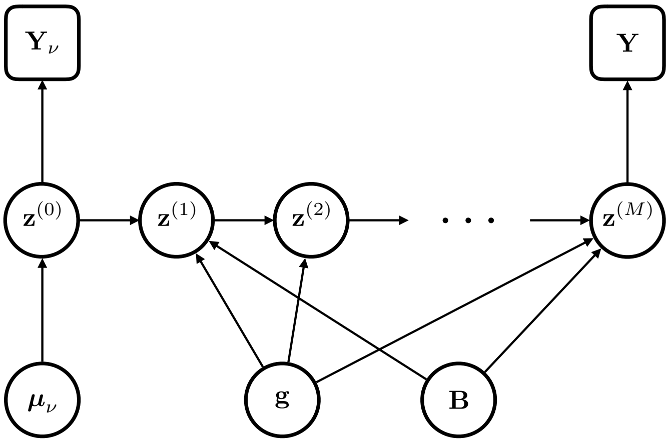


Figure 4.2: Graphical representation of the method of successive approximations model for the MLFM. The successive states $\mathbf{z}^{(m)}$ have the form of a simple linear Gaussian dynamic system. The data informs the distribution a two places; through the value of the initial approximation at time t_ν and the final output of the truncated successive approximations.

By considering the complete model with the set of latent states we shall be able to exploit the linear constraint conditions imposed by the Picard map. As a result we shall see that the log posteriors for the latent force and connection variables are defined by certain collections of quadratics leading to Gaussian conditionals analogous to the MLFM-AG approximation.

Using the conditional independence structure of the model, represented graphically in Figure 4.2, and the joint density function of the latent state variables (4.30) then

$$\begin{aligned} p(\mathbf{g} \mid \mathbf{Y}, \mathbf{Z}, \mathbf{B}, \boldsymbol{\psi}, \boldsymbol{\Gamma}, \boldsymbol{\phi}) &= p(\mathbf{g} \mid \mathbf{Z}, \mathbf{B}, \boldsymbol{\Gamma}, \boldsymbol{\psi}) \\ &\propto p(\mathbf{Z} \mid \mathbf{g}, \mathbf{B}, \boldsymbol{\Gamma}) \prod_{r=1}^{R} p(\mathbf{g}_r \mid \psi_r) \\ &\propto \prod_{i=1}^{M} \mathcal{N}(\mathbf{z}^{(m)} \mid \mathbf{P}[\mathbf{g}, \mathbf{B}]\mathbf{z}^{(m-1)}, \boldsymbol{\Gamma}) \prod_{r=1}^{R} \mathcal{N}(\mathbf{g}_r \mid \mathbf{0}, \mathbf{C}_{\psi_r}), \end{aligned} \tag{4.34}$$

which given that the entries of $\mathbf{P}[\mathbf{g}, \mathbf{B}]$ are linear in the forces will be a product of Gaussian densities, and so itself a Gaussian density. Before we provide an explicit representation of this density it is worth considering the interpretation discussed previous of $\boldsymbol{\Gamma}$ as a regularisation parameter and considering the limit $\boldsymbol{\Gamma} \rightarrow \mathbf{0}$, in which case we can, at least heuristicly, write the density (4.34) as

$$p(\mathbf{g} \mid \mathbf{Z}, \mathbf{B}, \boldsymbol{\phi}, \boldsymbol{\psi}) \propto \prod_{i=1}^{M} \delta\left(\mathbf{z}^{(m)} - \mathbf{P}[\mathbf{g}, \mathbf{B}]\mathbf{z}^{(m-1)}\right) p(\mathbf{g}), \tag{4.35}$$

where $p(\mathbf{g})$ is the Gaussian prior of the latent forces. So that the posterior acts to constrain the Gaussian prior onto the intersection of M linear subspaces determined by the solution constraints of the set of successive approximations. In this way the regularising term $\boldsymbol{\Gamma}$ inflates the posterior around this degenerate limit.

To derive the mean and covariance explicitly we first consider a given $m \in \{1, \ldots, M\}$, and for convenience letting $\mathbf{P} := \mathbf{P}[\mathbf{g}, \mathbf{B}]$, then

$$\begin{aligned} \log \mathcal{N}(\mathbf{z}^{(m)} \mid \mathbf{P}\mathbf{z}^{(m-1)}, \boldsymbol{\Gamma}) &= -\frac{1}{2}(\mathbf{P}\mathbf{z}^{(m-1)} - \mathbf{z}^{(m)})^{\top}\boldsymbol{\Gamma}^{-1}(\mathbf{P}\mathbf{z}^{(m-1)} - \mathbf{z}^{(m)}) + \text{const.}, \\ &= -\frac{1}{2}\operatorname{Tr}\left((\boldsymbol{\Gamma}^{-1}\mathbf{P}\mathbf{z}^{(m-1)})^{\top}\mathbf{P}\mathbf{z}^{(m-1)}\right) \\ &\quad + \operatorname{Tr}\left((\boldsymbol{\Gamma}^{-1}\mathbf{z}^{(m)}\mathbf{z}^{(m-1)\top})^{\top}\mathbf{P}\right) + \text{const.} \\ &= -\frac{1}{2}\operatorname{vec}(\mathbf{P})^{\top}\left(\mathbf{z}^{(m-1)}\mathbf{z}^{(m-1)\top} \otimes \boldsymbol{\Gamma}^{-1}\right)\operatorname{vec}(\mathbf{P}) \\ &\quad + \operatorname{vec}(\boldsymbol{\Gamma}^{-1}\mathbf{z}^{(m)}\mathbf{z}^{(m-1)\top})^{\top}\operatorname{vec}(\mathbf{P}) + \text{const.} \\ &= -\frac{1}{2}\mathbf{g}^{\top}\mathbf{V}\left(\mathbf{z}^{(m-1)}\mathbf{z}^{(m-1)\top} \otimes \boldsymbol{\Gamma}^{-1}\right)\mathbf{V}\mathbf{g} \\ &\quad + \left[\operatorname{vec}(\boldsymbol{\Gamma}^{-1}\mathbf{z}^{(m)}\mathbf{z}^{(m-1)\top})^{\top} - \mathbf{v}_0^{\top}\left(\mathbf{z}^{(m-1)}\mathbf{z}^{(m-1)\top} \otimes \boldsymbol{\Gamma}^{-1}\right)\right]\mathbf{V}\mathbf{g} \\ &\quad + \text{const.}, \end{aligned} \tag{4.36}$$

where the pair $(\mathbf{V}, \mathbf{v}_0)$ such that $\mathbf{V}\mathbf{g} + \mathbf{v}_0 = \operatorname{vec}(\mathbf{P})$ was defined by (4.21). In what follows it will be useful to define the statistics

$$\boldsymbol{\Psi}_0 = \sum_{m=1}^{M} \mathbf{z}^{(m-1)}\mathbf{z}^{(m-1)\top}, \qquad \boldsymbol{\Psi}_1 = \sum_{m=1}^{M} \mathbf{z}^{(m-1)}\mathbf{z}^{(m)\top}, \tag{4.37}$$

then taking the log transform of the density (4.34), using (4.36) and identifying coefficients then we may conclude that the conditional distribution of $\mathbf{g}$ is given by

$$p(\mathbf{g} \mid \mathbf{Z}, \mathbf{B}, \boldsymbol{\Gamma}, \boldsymbol{\psi}) = \mathcal{N}\left(\mathbf{g} \mid \mathbf{m}_g, \mathbf{K}_g\right), \tag{4.38}$$

where

$$\mathbf{m}_g(\mathbf{Z}, \mathbf{B}, \boldsymbol{\Gamma}, \boldsymbol{\psi}) = \mathbf{K}_g \mathbf{V}^\top \left[\operatorname{vec}\left(\boldsymbol{\Gamma}^{-1}\boldsymbol{\Psi}_1\right) - \left(\boldsymbol{\Psi}_0 \otimes \boldsymbol{\Gamma}^{-1}\right)\mathbf{v}_0\right] \tag{4.39a}$$

$$\mathbf{K}_g(\mathbf{Z}, \mathbf{B}, \boldsymbol{\Gamma}, \boldsymbol{\psi}) = \left(\mathbf{V}^\top\left(\boldsymbol{\Psi}_0 \otimes \boldsymbol{\Gamma}^{-1}\right)\mathbf{V} + \mathbf{C}_{\boldsymbol{\psi}}^{-1}\right)^{-1}. \tag{4.39b}$$

A direct interpretation of these moments is less clear than the heuristic density (4.35), but when we consider a parameterisation $\boldsymbol{\Gamma}^{-1} = \lambda \mathbf{I}_{NK}$ then the maximum of the conditioned posterior can be interpreted as the Tikhonov regularised solution.

Posterior conditional for the connection coefficients

We derive the conditional distribution for the connection coefficients, $\mathbf{B}$, using the same method as just considered for the latent forces. For the connection variables the conditional independence structure leads to the unnormalised posterior

$$\begin{aligned} p(\mathbf{B} \mid \mathbf{Y}, \mathbf{Z}, \mathbf{g}, \boldsymbol{\Gamma}, \boldsymbol{\phi}, \boldsymbol{\psi}, \boldsymbol{\zeta}) &= p(\mathbf{B} \mid \mathbf{Z}, \mathbf{g}, \boldsymbol{\Gamma}, \boldsymbol{\zeta}) \\ &\propto \prod_{m=1}^{M} \mathcal{N}(\mathbf{z}^{(m)} \mid \mathbf{P}[\mathbf{g}, \mathbf{B}]\mathbf{z}^{(m-1)}, \boldsymbol{\Gamma}) p(\mathbf{B} \mid \boldsymbol{\zeta}). \end{aligned}$$

Again this density may be interpreted as concentrating the prior around the intersection of linear spaces defined by the linear constraints of the successive approximations.

The rearrangement analogous to (4.36) is given by

$$\begin{aligned} \log \mathcal{N}(\mathbf{z}^{(m)} \mid \mathbf{P}\mathbf{z}^{(m-1)}, \boldsymbol{\Gamma}) &= -\frac{1}{2}\operatorname{vec}(\mathbf{P})^\top \left(\mathbf{z}^{(m-1)}\mathbf{z}^{(m-1)\top} \otimes \boldsymbol{\Gamma}^{-1}\right)\operatorname{vec}(\mathbf{P}) \\ &\quad + \operatorname{vec}(\boldsymbol{\Gamma}^{-1}\mathbf{z}^{(m)}\mathbf{z}^{(m-1)})^\top \operatorname{vec}(\mathbf{P}) + \text{const} \\ &= -\frac{1}{2}\boldsymbol{\beta}^\top \mathbf{W}^\top \left(\mathbf{z}^{(m-1)}\mathbf{z}^{(m-1)\top} \otimes \boldsymbol{\Gamma}^{-1}\right)\mathbf{W}\boldsymbol{\beta} \\ &\quad + \left[\operatorname{vec}\left(\boldsymbol{\Gamma}^{-1}\mathbf{z}^{(m)}\mathbf{z}^{(m-1)\top}\right)^\top - \mathbf{w}_0^\top\left(\mathbf{z}^{(m-1)}\mathbf{z}^{(m-1)\top} \otimes \boldsymbol{\Gamma}^{-1}\right)\right]\mathbf{W}\boldsymbol{\beta} \\ &\quad + \text{const.}, \end{aligned} \tag{4.40}$$

for $m = 1, \ldots, M$. If we now use the representation of vec($\mathbf{P}$) in terms of its Jacobian, $\mathbf{W}$,taken with respect to the variables $\mathbf{g}$, and offset in the representation (4.21). Then using the Markov structure to sum over the contribution from each of the Gaussian transition densities we again realise a Gaussian conditional density of the form

$$p(\mathbf{B} \mid \mathbf{Z}, \mathbf{g}, \boldsymbol{\Gamma}, \boldsymbol{\zeta}) = \mathcal{N}(\boldsymbol{\beta} \mid \mathbf{m}_\beta, \mathbf{K}_\beta),$$

where using the statistics (4.37) the moments are given by

$$\mathbf{m}_\beta(\mathbf{Z}, \mathbf{g}, \boldsymbol{\Gamma}, \boldsymbol{\zeta}) = \mathbf{K}_\beta \mathbf{W}^\top \left[\operatorname{vec}\left(\boldsymbol{\Gamma}^{-1}\boldsymbol{\Psi}_1\right) - \left(\boldsymbol{\Psi}_0 \otimes \boldsymbol{\Gamma}^{-1}\right)\mathbf{w}_0\right], \tag{4.41a}$$

$$\mathbf{K}_\beta(\mathbf{Z}, \mathbf{g}, \boldsymbol{\Gamma}, \boldsymbol{\zeta}) = \left(\boldsymbol{\Psi}_0 \otimes \boldsymbol{\Gamma}^{-1} + \mathbf{C}_\zeta^{-1}\right)^{-1}. \tag{4.41b}$$

4.4 Marginalisation over the latent states

The introduction of the augmented latent variable set provides the model with the successive approximation structure which allowed for the construction of tractable conditions, but the state variables are of relatively little interest if our primary goal is to learn the coefficient matrix that governs this non-autonomous ODE. Furthermore, by including the latent state variables, we have a significant expansion in the size of the latent variable set. Indeed the $T \cdot K$ state variables in the MLFM-AG method have become $(M+1)\cdot N \cdot K$ variables with $N \geq T$, and possibly much larger if the data is sparse, but we desire a dense refinement of the interval over which we are solving the model.

Therefore in this section, we consider the possibility of alleviating this problem using the same marginalisation strategy as considered in Section 3.4. Indeed the strategy in the previous chapter may be summarised as introducing the gradient process to impose the model structure, marginalise over the gradient process, and then also use the linear structure to marginalise over the state process itself to lead to a parsimonious representation of the model ideal for forming point estimates. We follow a similar approach here where after having introduced the latent variables to approximate the series expansion solution to the trajectory, and so we now examine the possibility of marginalising out this augmented variable set leaving the residual model structure. In doing so we shall construct the marginal likelihood term $p(\mathbf{Y} \mid \mathbf{g}, \mathbf{B}) = \int p(\mathbf{Y}, \mathbf{Z} \mid \mathbf{g}, \mathbf{B})\mathrm{d}\mathbf{Z}$, which will provide a more parsimonious parameterisation of the optimisation problems we are interested in.

The Markov structure of the conditional density of the latent variables $\{\mathbf{z}^{(m)}\}_{m=0}^{M}$ means that it is straightforward to construct the marginal density by integrating over (4.30) leading to

$$\begin{aligned}
p(\mathbf{Y}|\mathbf{g},\mathbf{B},\boldsymbol{\sigma},\boldsymbol{\Gamma},\boldsymbol{\phi}) &= \int p(\mathbf{Y} \mid \mathbf{Z}^{(M)}, \boldsymbol{\sigma}) p(\mathbf{Z} \mid \mathbf{g}, \mathbf{B}, \boldsymbol{\phi}, \boldsymbol{\Gamma}) \mathrm{d}\mathbf{Z} \\
&= \int \cdots \int \mathcal{N}(\mathbf{y} \mid \mathbf{z}^{(M)}, \operatorname{diag}(\boldsymbol{\sigma}^{\circ 2}) \otimes \mathbf{I}_N) \prod_{m=2}^{M} \mathcal{N}(\mathbf{z}^{(m)} \mid \mathbf{P}[\mathbf{g},\mathbf{B}]\mathbf{z}^{(m-1)}, \boldsymbol{\Gamma}) \\
&\qquad \left(\int \mathcal{N}(\mathbf{z}^{(1)} \mid \mathbf{P}[\mathbf{g},\mathbf{B}]\mathbf{z}^{(0)}, \boldsymbol{\Gamma}) \mathcal{N}(\mathbf{z}^{(0)} \mid \mathbf{0}, \mathbf{C}_0) \mathrm{d}\mathbf{z}^{(0)}\right) \mathrm{d}\mathbf{z}^{(1)} \cdots \mathrm{d}\mathbf{z}^{(M)} \\
&= \int \cdots \int \mathcal{N}(\mathbf{y} \mid \mathbf{z}^{(M)}, \operatorname{diag}(\boldsymbol{\sigma}^{\circ 2}) \otimes \mathbf{I}_N) \prod_{m=2}^{M} \mathcal{N}(\mathbf{z}^{(m)} \mid \mathbf{P}[\mathbf{g},\mathbf{B}]\mathbf{z}^{(m-1)}, \boldsymbol{\Gamma}) \\
&\qquad \times \mathcal{N}(\mathbf{z}^{(1)} \mid \mathbf{0}, \mathbf{P}\mathbf{C}_0\mathbf{P}^\top + \boldsymbol{\Gamma}) \mathrm{d}\mathbf{z}^{(1)} \cdots \mathrm{d}\mathbf{z}^{(M)} \\
&= \mathcal{N}(\mathbf{x}|\mathbf{0}, \mathbf{C}_M + \operatorname{diag}(\boldsymbol{\sigma}^{\circ 2}) \otimes \mathbf{I}_N),
\end{aligned} \tag{4.42}$$

where $\mathbf{C}_M = \mathbf{C}_M(\mathbf{g}, \mathbf{B}, \boldsymbol{\Gamma}, \boldsymbol{\phi})$ is defined recursively by

$$\mathbf{C}_m = \mathbf{P}[\mathbf{g},\mathbf{B}]\mathbf{C}_{m-1}\mathbf{P}[\mathbf{g},\mathbf{B}]^\top + \boldsymbol{\Gamma}, \tag{4.43}$$

and the initial covariance matrix $\mathbf{C}_0$ is given by the covariance matrix of $\mathbf{Z}^{(0)}$.

To interpret this result, it is again useful to consider the deterministic limit. If we also consider the case where we are using a constant approximation to the initial state, then the initial covariance matrix is given by

$$\mathbf{C}_0 = \sum_{k=1}^{K} \phi_k^{-1} (\mathbf{e}_k \otimes \mathbf{1}_N)(\mathbf{e}_k \otimes \mathbf{1}_N)^\top, \tag{4.44}$$

where $\mathbf{e}_k$ is the standard basis vector in $\mathbb{R}^K$. Repeated application of (4.43) with $\mathbf{\Gamma} = 0$ leads to

$$\mathbf{C}_M = \sum_{k=1}^{K} \phi_k^{-1} \left[\mathbf{P}^M(\mathbf{e}_k \otimes \mathbf{1}_N)\right] \left[\mathbf{P}^M(\mathbf{e}_k \otimes \mathbf{1}_N)\right]^\top. \tag{4.45}$$

It follows that in the deterministic case the covariance matrix is singular with rank of at most K. In the limit $M \to \infty$ the vector $\mathbf{P}^M(\mathbf{e}_k \otimes \mathbf{1}_N)$ is a discrete approximation to the IVP

$$\dot{\mathbf{x}}(t) = \mathbf{A}(t)\mathbf{x}(t), \qquad \mathbf{x}(t_0) = \mathbf{e}_k,$$

for $k = 1, \ldots, K$, so that we may directly relate the covariance matrix to the fundamental solution of the ODE problem. This reflects our attempts in this chapter to construct an approximation to the distribution which is more faithful to the ODE model as a transformation of the underlying latent forces analogous to that presented in Section 2.2.1 for the LFM, rather than the use of the interpolating processes in the MLFM-AG which sidesteps this feature. Unfortunately, unlike for the LFM in which the linear transformation acts on the latent forces we now have a situation in which the linear transformation depends directly on the latent forces and so integrating over the force variables is not possible.

Returning to the general case with nonzero regularisation matrix the result is the density of a mean zero Gaussian random variable where the covariance matrix is a degree $2M$ polynomial in the latent force variables and coefficient variables, contrast this with the covariance matrix for the adaptive gradient matching model (3.45) in which the inverse covariance matrix was a quadratic in the latent force variables. Given this, potentially high degree, polynomial dependence of the covariance matrix and further compounded by the fact that representing the log-likelihood will require inverting this covariance matrix, then methods based on maximising the marginal-likelihood term (4.42) will necessarily be computationally more burdensome than the analogous methods discussed for the adaptive gradient matching methods, although it is, in principle, possible to construct the gradient of the covariance matrix, $\mathbf{C}_M$, with respect to the model parameters and so use gradient-based optimisation methods but the recursive structure of the covariance matrix makes this an expensive process. However, simulation studies presented in Chapter 6 demonstrate that there are situations where this increased computational burden is rewarded with more accurate solutions, and so in the next chapter we will discuss an expectation maximisation (EM) method for reducing some of the computational complexity, and importantly allow for gradient-free optimisation. In particular, the variational methods in the next chapter will reintroduce the "collection of quadratics" structure that made deriving the conditional distributions above possible, and so by making a more principled use of the linear dynamic systems form of the complete data likelihood, we will regain some of the computational efficiency that is lost after marginalisation.

As well as the consideration of the point estimates by optimisation of the likelihood term (4.42) we will also be interested in obtaining estimates of the posterior distributions of our variables of interest. Doing so will necessarily involve using Bayes rule to invert the likelihood term (4.42), but the complex, nonlinear dependence of the covariance matrix is going to make performing this process infeasible, and again we shall see that the introduction of variational methods allow us to construct efficient approximations.

4.4.1 Feed-forward successive approximations

Before proceeding we discuss one further interpretation of the model in the deterministic setting with $\mathbf{\Gamma} = 0$. In this instance we no longer have the problematic augmentation of our variable set — the variables $\{\mathbf{z}^{(m)}\}_{m=1}^{M}$ are the successive outputs formed by applying the Picard operator starting from some initial condition rather than free variables to be optimised.

If we consider the case with initial input given by the constant inital approximation $\mu \in \mathbb{R}^K$ then the conditional density of the outputs is given by

$$p(\mathbf{Y} \mid \mathbf{g}, \mathbf{B}, \boldsymbol{\mu}) = \mathcal{N}\left(\mathbf{y} \mid \mathbf{P}^M \left(\mathbf{1}_N \otimes \boldsymbol{\mu}\right), \operatorname{diag}(\boldsymbol{\sigma}) \otimes \mathbf{I}_N\right). \tag{4.46}$$

If we were to further marginalise over $\boldsymbol{\mu}$ then we would recover the intractable density (4.42) with $\mathbf{\Gamma} = \mathbf{0}$. Instead we retain these variables as additional parameters to be optimised.

Each set of the latent variables $\mathbf{z}^{(m)}$, $m = 1, \ldots, M$ is formed as a linear combination of the preceding variables, and as such may be identified as a particular case of the feed-forward neural network architecture [Bishop, 1995, 2006]. For a general network, the transformation between layers is given by

$$\mathbf{z}_i^{(m)} = f\left(\sum_j c_{ij}\mathbf{z}_j^{(m-1)} + b_i\right),$$

for a collection of scalar weights $\{c_{ij}\}$, biases b_j, and some function, f, referred to as the activation function. In our case we have

$$\mathbf{z}_i^{(m)} = [\mathbf{K}[\mathbf{g}, \mathbf{B}]\mathbf{z}^{(m-1)}]_i + \mathbf{z}_0,$$

where $\mathbf{K}[\mathbf{g}, \mathbf{B}]$ is the discretised operator (4.20). Therefore the activation function is given by the identity function, the biases set equal to the initial conditions, and the weights corresponding to the entries of the matrix $\mathbf{K}$, that is

$$c_{ij} = (\mathbf{P}[\mathbf{g}, \mathbf{B}])_{ij}.$$

The input of this neural network will be the random variables, μ_k, representing the initial values as described in Section 4.3.3. These then get transformed to the first layer through the mapping

$$\mathbf{z}_k^{(0)} = \mathbf{1}_N \cdot \mu_k, k = 1, \ldots, K.$$

This allows us to view the deterministic case as a feed-forward neural network with weight parameters shared between layers. From a conceptual point of view, this adds

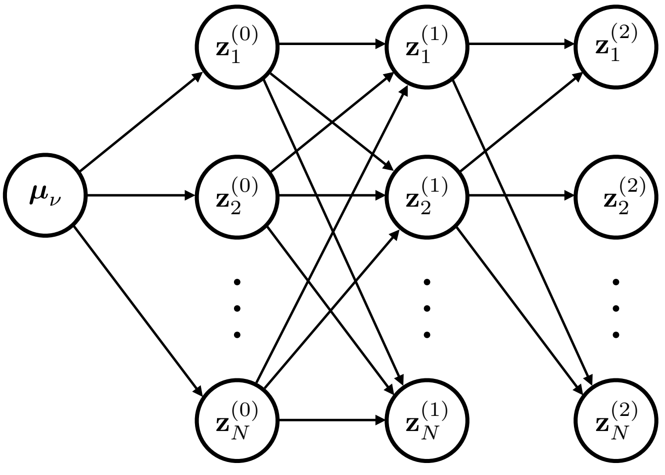


Figure 4.3: Feed forward structure of the deterministic transition MLFM-SA.

little to the interpretation of the model. However, the interpretation of the method of successive approximations as a neural network has at least two significant benefits. The first benefit of the neural network architecture is its prevalence in modern machine learning, and so this interpretation allows the MLFM-SA method to be easily incorporated into more complex inference architectures. Secondly, the use of backpropagation to evaluate the gradients of neural networks with respect to their model parameters. The current popularity of neural networks has to lead to the existence of many efficient implementations of this process such as in the TensorFlow [Abadi et al., 2015] software package, allow one to adapt existing implementations to allow for straightforward calculation of the gradients using automatic differentiation methods. Increasing, or decreasing, the approximation order could also be viewed as a problem of selecting the network topology allowing existing methods for handling this problem to be adapted to our setting.

One of the possible shortcomings of the MLFM-SA approximation is the fact that a given approximation is only locally accurate. The neural network interpretation of the MLFM-SA methods suggests the possibility of addressing this shortcoming by utilising existing network architectures designed to model this sort of locality, as an example we could consider using convolution layers [Le Cun et al., 1989, 1998] to model decay in the accuracy of the approximation as the time interval increases. We do not discuss this particular extension in this thesis; instead, we introduce a mixture modelling framework in the next section to address the same shortcoming, but it is suggestive of the possibilities of adapting the MLFM-SA method using the neural network architectures and a possible avenue for future research.

4.5 Mixtures of the successive approximation model

The construction of MLFM-SA model in this chapter involves fixing an arbitrary initial point, and while the actual value of the state at this fixed point may be integrated out, the effect of this is still embodied in the construction of the discretised Picard operator (4.20) which when acting on a vector leaves some $\mathbb{R}^K$ sub-vector fixed.

Since a single application of this operator, ignoring the quadrature error, is only accurate up to terms of $\mathcal{O}(|t - t_0|^2)$ it is reasonable to expect that for large sample intervals we are going to need to take increasingly more iterations of the Picard map to get an accurate approximate over the whole interval. However, as we have discussed in Section 4.4, the process of carrying out joint inference over the augmented latent state variable set will quickly become infeasible as the order of approximation increases.

In this section we address this issue by first considering several local approximations to the trajectory, each of lower order and with a distinct initial time. These are then combined to produce an alternative to a single approximation of higher order. Because of the polynomial structure of the covariance matrix (4.43) the additive combination of several lower order methods will often have a lower computational complexity than a single higher order method.

To carry out this construction we consider a collection of, Q, distinct fixed indices with corresponding initial times $\{\tau_{\nu_q}\}_{q=1}^{Q}$. To each choice of initial time there will also be a corresponding initial value, and so we also introduce a collection of initial latent states, $\mathbf{Z}^{(q,0)}$, and the family of successive approximations obtained by applying the operator $\mathbf{P}_q$ to each of these initial trajectories. We shall denote the complete collection of different versions of the successive approximations by $\mathbf{Z}^{(q)} = \{\mathbf{Z}^{(q,0)}, \ldots \mathbf{Z}^{(q,M)}\}$, for $q = 1, \ldots, Q$. In principle we could also consider allowing a different order, M_q, for each family of approximations although that is not a development we consider here.

Again recalling our remarks in the previous section we note that any concerns about the size of the parameter vector to be estimated in the case of a single component are going to be further compounded in the setting currently proposed. We ignore this issue for now, but will return to this point in Chapter 5 when we demonstrate the possibility of marginalising over the latent state approximations inside a variational scheme.

Each set of approximations will be accurate around its local initial value, and increasingly accurate over the whole interval as the approximation order increases. Intuitively we view each collection of variables as having been generated by a 'local expert' and these local estimators are then to be combined in order to produce an accurate global model. A representation of this construction is given in Figure 4.4 for two such local experts centred at τ_1 and τ_2, represented in Figure 4.4a and Figure 4.4b respectively. Around their initial conditions these experts are well centred at the true trajectory, but the accuracy decays as the time interval increases, we stress that there is no reason to expect this decay to be monotone. In Figure 4.4c we represent graphically the process of combining these local estimators to form a single estimator with a superior performance over the whole interval.

Conditional on the parameters each local model will have exactly the form of the Markov chain density constructed in Section 4.3.3, and so each local expert is represented by a conditional density given by

$$\begin{aligned} p_q(\mathbf{Y}, \mathbf{Z}^{(q)} \mid \mathbf{g}, \mathbf{B}, \boldsymbol{\sigma}, \boldsymbol{\Gamma}, \boldsymbol{\phi}) &= p(\mathbf{y} \mid \mathbf{z}^{(q,M)}, \boldsymbol{\sigma}) p(\mathbf{z}^{(q,0)} \mid \boldsymbol{\phi}) p(\mathbf{z}^{(q,1)}, \ldots, \mathbf{z}^{(q,M)} \mid \mathbf{z}^{(q,0)}, \mathbf{g}, \mathbf{B}, \boldsymbol{\Gamma}) \\ &= \mathcal{N}(\mathbf{y} \mid \mathbf{z}^{(q,0)}) \mathcal{N}(\mathbf{z}^{(q,0)} \mid \mathbf{0}, \mathbf{C}_{\boldsymbol{\phi}}) \prod_{m=1}^{M} \mathcal{N}(\mathbf{z}^{(q,m)} \mid \mathbf{P}_q[\mathbf{g}, \boldsymbol{\beta}] \mathbf{z}^{(q,m-1)}, \boldsymbol{\Gamma}), \end{aligned} \tag{4.47}$$

for $q = 1, \ldots, Q$. Similarly the corresponding marginalised version of each expert is obtained following the process described in Section 4.4 and so will have a conditional density given by

$$\begin{aligned} p_\nu(\mathbf{Y} \mid \mathbf{z}^{(\nu,0)}, \mathbf{g}, \mathbf{B}, \mathbf{\Gamma}) &= \int p_\nu(\mathbf{Y}, \mathbf{Z}^{(\nu)} \mid \mathbf{z}^{(\nu,0)}, \mathbf{g}, \mathbf{B}, \mathbf{\Gamma}) \mathrm{d}\mathbf{Z}^{(\nu)} \\ &= \mathcal{N}(\mathbf{y} \mid \mathbf{0}, \mathbf{C}_{\nu,M} + \operatorname{diag}(\boldsymbol{\sigma}^{\circ 2}) \otimes \mathbf{I}_N), \end{aligned} \tag{4.48}$$

where the covariance matrix $\mathbf{C}_{\nu,M}$ is again given by the recursive construction

$$\mathbf{C}_{\nu,m} = \mathbf{P}_\nu \mathbf{C}_{\nu,m-1} \mathbf{P}_\nu^\top + \mathbf{\Gamma}, \qquad m = 1, \ldots, M,$$

and $\mathbf{C}_{\nu,0} = \mathbf{C}_0$, in principal we could further allow expert specific initial covariance matrices, but this increases the number of model parameters for no obvious benefit. Note that each of the operators, $\mathbf{P}_\nu$, shares the common set of model parameters $\{\mathbf{g}, \mathbf{B}\}$ and so differs only in the specification of the quadrature weight matrix in (4.14), and the index of the point to be left fixed.

To complete the model we would like to combine each different version of the approximating process to construct a single approximation to the density of the trajectory by combining the local expert models. A natural way of doing this is to combine them as a Gaussian process mixture model. Therefore we specify a D-vector, $\boldsymbol{\pi}$, of mixture component probabilities with $\sum_{\nu=1}^{D} \pi_\nu = 1$ and $\pi_\nu \geq 0$ for $\nu = 1, \ldots, D$. The resulting approximation to the density is given by

$$p(\mathbf{y} \mid \mathbf{g}, \mathbf{B}, \boldsymbol{\phi}, \mathbf{\Gamma}) = \sum_{\nu=1}^{D} \pi_\nu p_\nu(\mathbf{Y} \mid \mathbf{g}, \mathbf{B}, \mathbf{\Gamma}, \boldsymbol{\sigma}) \tag{4.49}$$

with each mixture component density given by (4.48). The graphical representation of this model is given in Figure 4.4d, where we have used the plate notation [Buntine, 1994] to represent the conditional independence assumption

$$p(\mathbf{Z}^{(1)}, \ldots, \mathbf{Z}^{(\nu)} \mid \mathbf{g}, \mathbf{B}, \boldsymbol{\phi}, \mathbf{\Gamma}) = \prod_{\nu=1}^{D} p(\mathbf{Z}^{(\nu)} \mid \mathbf{g}, \mathbf{B}, \boldsymbol{\phi}, \mathbf{\Gamma}).$$

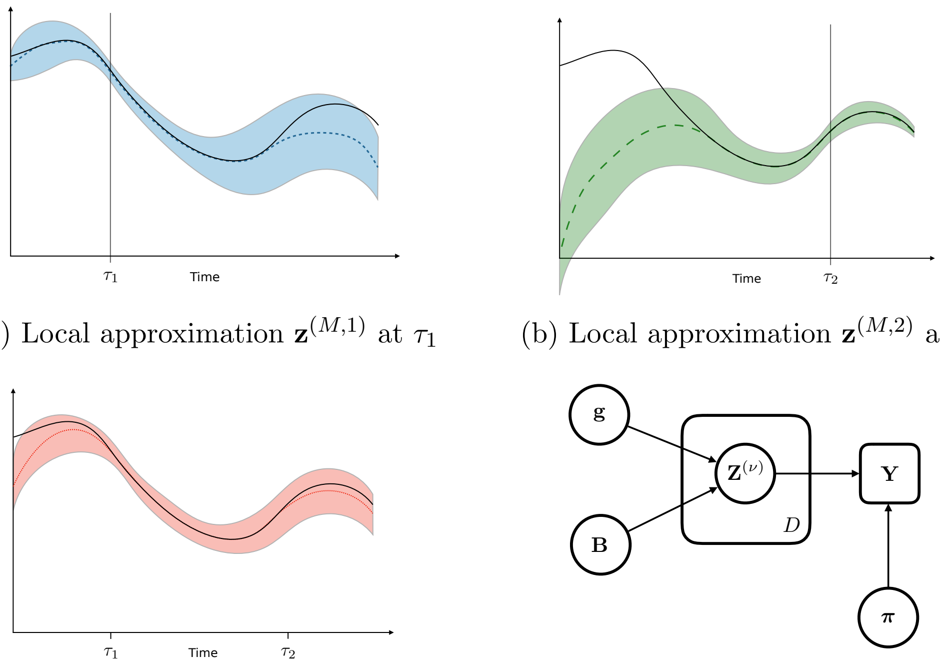
Combining the local models as a mixture distribution constant mixing coefficients $\boldsymbol{\pi}$ leads to a tractable Gaussian mixture model. However, it is perhaps an overly simplistic form of model combination, and it may be possible to consider more elaborate methods of combining these local approximations to the density. One possible alternative might be to consider allowing the mixture coefficients to depend on the input time leading to a hierarchical mixture of experts [Jordan and Jacobs, 1994].

A simple example of a model with the mixture probability depending on the input time is given by combining local versions of the deterministic density function (4.46) discussed in Section 4.4.1. Leading to a mixture density

$$p(\mathbf{y}_i \mid \mathbf{g}, \mathbf{B}, \boldsymbol{\mu}) = \sum_{\nu=1}^{D} \pi_\nu(t_i) \mathcal{N}(\mathbf{y} \mid (\mathbf{P}_\nu[\mathbf{g}, \mathbf{M}]^M (\mathbf{1}_N \otimes \boldsymbol{\mu}_\nu))_i, \sigma^2), \tag{4.50}$$

with the mixing coefficients given by softmax functions of the form

$$\pi_\nu(t) = \frac{e^{h(t - t_\nu)}}{\sum_{\nu'=1}^{Q} e^{h(t - t_{\nu'})}}, \tag{4.51}$$



(a) Local approximation $\mathbf{z}^{(M,1)}$ at τ_1

(b) Local approximation $\mathbf{z}^{(M,2)}$ at τ_2

(c) Combination of local approximations

(d) MLFM-MixSA network graph

Figure 4.4: Representation of the construction of the MLFM-MixSA model. The true trajectory is represented in (a), (b) and (c) by '—'. The conditional independence structure of the model is represented in (d).

for some choice of basis functions h. This would then allow us to directly model the local nature of the MLFM successive approximations, by for instance choosing the basis functions so that the probability of belonging to a given mixture decays away from the corresponding initial time.

It is not immediately clear whether the increased modelling complexity will provide any additional benefits, but an examination of this issue, perhaps in conjuction with the idea of, temporal, convolution neural networks as mentioned at the end of the preceeding section may be a worthwhile area of study.

4.6 Discussion

In this chapter, we have introduced the method of successive approximations to the solutions of ODEs and used it to motivate an approximate solution for the MLFM. For the remainder of this work, we refer to this as the MLFM successive approximation method (MLFM-SA). The model is necessarily an approximation, as was MLFM-AG method of the previous chapter, and balances the tractability of the linear dynamic system approximation to the set of successive approximations, with the computational complexity of increasing the resolution of the trajectory. It is an important area of future work to establish more rigorous mathematical results concerning this approximation, and we discuss this issue again in our final chapter.

Our method works by treating the set of successive approximations as the "complete data" for our model, and conditioning on these variables we can present tractable conditional distributions in Section 4.3.3. A given pair of successive approximations act to constrain the parameters to certain linear subspaces of the full parameter space, and the resulting conditionals of the coefficient matrix take the form of Gaussians concentrated around the intersection of these hyperplanes.

Unfortunately, this process of model completion vastly expands the size of the variable set, making joint optimisation implausible. Moreover, while we demonstrated in Section 4.4 that it is possible to marginalise over the complete data model the resulting distribution is no longer tractable. The sum of M quadratics gets replaced by a covariance matrix, the entries of which are degree $2M$ polynomials in the latent forces. This distribution is not only intractable for calculating an analytic posterior but also computationally expensive and numerically unstable when calculating gradients as part of an optimisation procedure. In the next chapter, we demonstrate that by using variational methods we can realise techniques that manage to combine both the marginalisation step with the more appealing structure of the full data model.

While we have produced conditional distributions for all of the model parameters we have consistently stressed that the covariance matrix in the transition distribution of the LDS model for the successive approximations should be viewed as regularisation term. This parameter is likely to be very small which makes updating the conditional distribution of the state variables inside an MCMC routine harder. The efficiency of the LDS comes from the Kalman filtering and smoothing equations which avoids us ever having to construct the full moments of this distribution. Instead, we only need specific statistics involving the marginal and pairwise distributions. Unfortunately, because of the very small covariance term between approximations, any form of update using only adjacent approximations will be very small. As such we do not recommend using the model introduced in this chapter as part of an MCMC routine, and instead, find it more appropriate as part of the variational approximation which we introduce in the next chapter. When we set this regularisation term to be the zero matrix so that the transitions are deterministic, we arrive at the feed-forward structure discussed in Section 4.4.1.

Both the MLFM-AG method of the previous chapter and the MLFM-SA method introduced in this chapter have required the introduction of hyperparameters and regularising parameters. In this chapter, we introduced the parameters $\boldsymbol{\phi}$ to control the scale of the initial approximation to the solution, and the parameter $\boldsymbol{\Gamma}$ to control the transition between successive approximations. Unlike for the MLFM-AG method, these parameters have much better conditional independence structures, and importantly they do not get transformed inside the conditional densities by complex nonlinear kernel function or matrix inverse operations.

A feature of our use of the integral transform (4.6) to construct solutions to the model is that the transformation between approximations fixes a point and that the accuracy of the approximation decays with the distance from the initial time. To address this, we discussed in Section 4.5 of combining multiple local estimators of the model to construct a better global approximation. In this chapter we have considered using a mixture model to carry out this task, we discuss an alternative approach based on the idea of local polynomial regression in Section 8.2.2.

We have now introduced two approximations to our MLFM, both of which attempt to use certain fixed point conditions and regularisation parameters to embed the ODE model structure into a more general approximation. For the MLFM-AG model this was by convolving a regression model for the time-evolution equation with a GP prior on the state variables, and for the MLFM-SA model, this was achieved by approximating a series expansion to the solution of the model in integral form as an LDS. In their current form the MLFM-AG model is a much more parsimonious parameterisation of the problem, and so in the next chapter we discuss variational methods that allow for practical fitting of the MLFM-SA method, and in Chapter 6 we demonstrate that the MLFM-SA method can perform well in situations that the MLFM-AG method fails.

Chapter 5

Variational inference for the multiplicative latent force model

5.1 Introduction

Our work in the previous chapters has introduced two different approaches to motivating a joint density for the variables in our MLFM. An important feature of both methods is the presence of an augmented variable set such that for the complete set of variables we can take advantage of a tractable conditional Gaussian structure, but that upon marginalisation over the augmented variables this structure is lost.

In each case, the relevant augmented variables were the latent trajectory variables, or more accurately in the case of the successive approximation method a truncated set of approximations to this variable. If our interest is in deriving point estimates of the coefficient matrix of the MLFM, which governs the dynamics, then any joint optimisation with respect to this much larger augmented variable set would be inefficient. We therefore also considered the marginal likelihood terms in Section 3.4 and Section 4.4 obtained after marginalisation over the trajectory variables, this then leaves a smaller set of variables for which the joint density is be optimised with respect to. In both cases the tractable posterior conditionals for the variables $\mathbf{g}$ and $\mathbf{B}$ are replaced with a Gaussian likelihood term of the form $p(\mathbf{Y} \mid \mathbf{g}, \mathbf{B})$ for the observations conditional on these parameters, but in both cases a likelihood in which the covariance matrix has a nonlinear dependence on these parameters. This complex nonlinear dependence was particularly pronounced for the MLFM-SA model in which the covariance matrix became a, potentially high degree, polynomial in the latent force and coefficient variables, and so for this model, we do not regard the direct optimisation of the likelihood as being a viable strategy for this class of models. The presence of an augmented latent variable set that allows for a tractable conditional independence structure suggests seeking alternative methods of inference which better respect this model structure, rather than performing the marginalisation step which obscures this structure. Once such example is the expectation maximisation (EM) algorithm which is an iterative procedure which replaces the single marginalisation over the state variables with an iterative procedure involving marginalisation of the state variables in the joint likelihood with respect to an updated version of the posterior conditional of the latent variables. For the MLFM-AG method, this process does not lead to a significant improvement over direct maximisation of the likelihood, but we shall see in this chapter that it leads to a significantly simplified optimisation routine for the MLFM-SA method.

As well as point estimates we will also be interested in deriving distributional estimates for which the same general point holds — for the full variable set we have the tractable conditional distributions presented in Section 3.3.1 and Section 4.3.3, but an intractable marginalised posterior conditional. While properly implemented Monte Carlo methods have the benefit of guaranteeing an eventual convergence towards sampling from the sought after posterior this rate of convergence may be slow, and as such it may be of interest to consider deterministic methods for approximating the unknown distribution. Variational inference, of which the EM algorithm can be regarded as a special case, is one such class of deterministic approximations introduced by minimising a functional involving the true density and members of a, simpler, approximating class using variational methods. We shall be particularly interested in the mean-field variational approach, which attempts to approximate the true distribution with the optimal product of independent factors. This method is closely connected to Gibbs sampling, and in those cases that the model possesses a straightforward Gibbs sampling routine, it will often have an easy to implement mean-field approximation, where each tractable conditional in the Gibbs sampling routine is closely related to a factor of the mean field approximation.

In this chapter, we first present the EM approach to finding optimal estimates of the parameters in the latent force model. We shall discuss the case of both the MLFM-AG and the MLFM-SA model, although it is in the latter case which required a more extensive augmenting variable set that the most significant benefit is observed. We then briefly discuss the more general variational inference framework with a focus on mean field methods and derive the resulting mean field variational approximations to both the MLFM-AG and the MLFM-SA methods introduced in this thesis.

5.2 Expectation maximisation

The expectation maximisation, or EM, algorithm [Dempster et al., 1977, McLachlan and Krishnan, 1997] provides a practical framework for carrying out either maximum likelihood or maximum a posteriori inference for probabilistic models with latent variables, as well as providing an important conceptual step in the extension to the more general variational inference framework which we will demonstrate in the next section. For the case of the MLFM model, we are principally interested in performing MAP estimation of the coefficient matrix $\mathbf{A}(t)$ which is comprised of the latent forcing functions and connection coefficient parameters. In some situations, we shall also be interested in providing an estimate of the latent state itself, and the scale of the observation noise. The remaining hyperparameters which define the kernel function of the introduced GPs, either the latent forces common to both methods or the state interpolants in the case of the MLFM-AG matching method, we shall regard as nuisance parameters.

In the approximate models introduced in Chapter 3 and Chapter 4 we demonstrated that for both instances it was possible to marginalise over the latent state variables entirely, but in doing so, we produce likelihood functions that are complex functions of the latent force variables. This complexity was particularly pronounced for the MLFM-SA model described in Section 4.3 where the covariance matrix of the marginal likelihood (4.30) was a degree $2M$ polynomial in the latent forces. If instead of marginalising we retained the latent states as a set of complete data then the resulting conditional distribution structure was much more amenable to inference, and the EM algorithm will provide us with the tools to carry out inference by making use of this appealing conditional structure. We first present a brief discussion of the general EM algorithm following [Bishop, 2006] before going on to show we can apply the algorithm to the adaptive gradient matching and successive approximations methods respectively.

In general, we shall be concerned with carrying out inference on some set of parameters $\boldsymbol{\theta}$ from a parameterised statistical model on the basis of having observed data $\mathbf{Y}$. We also assume there is some additional set of latent variables which we denote by $\mathbf{Z}$, and that these variables contribute non-trivial information to the complete joint likelihood term $p(\mathbf{X}, \mathbf{Y} \mid \boldsymbol{\theta})$. By marginalisation over the latent variables we have that the log-likelihood satisfies the inequality

$$\begin{aligned} \log p(\mathbf{Y} \mid \boldsymbol{\theta}) &= \log \left(\int_{\mathcal{Z}} p(\mathbf{Y}, \mathbf{Z} \mid \boldsymbol{\theta}) \mathrm{d}\mathbf{Z} \right) \\ &\geq \int_{\mathcal{Z}} q(\mathbf{Z}) \log \left(\frac{p(\mathbf{Y}, \mathbf{Z} \mid \boldsymbol{\theta})}{q(\mathbf{Z})} \right) \mathrm{d}\mathbf{Z}, \end{aligned} \tag{5.1}$$

for any choice of approximating distribution $q(\mathbf{Z})$ with support $\mathcal{Z}$. The EM algorithm works by taking the choice $q(\mathbf{Z}) = p(\mathbf{Z} \mid \mathbf{Y}, \boldsymbol{\theta}_{old})$ for some current value of the model parameters, $\boldsymbol{\theta}_{old}$, and iteratively updates the parameters by maximising the bound (5.1).

In general the EM algorithm is going to be useful in cases, such as those encountered in the previous chapters, where the joint distribution $p(\mathbf{Y}, \mathbf{Z} \mid \boldsymbol{\theta})$ is tractable, but the marginalised distribution $p(\mathbf{Y} \mid \boldsymbol{\theta})$ is not. We refer to the pair $\{\mathbf{Y}, \mathbf{Z}\}$ as the *complete* data set and $\log p(\mathbf{Y}, \mathbf{Z} \mid \boldsymbol{\theta})$ as the *complete data log likelihood* and we shall assume that the form of this is straightforward to work with. It is natural to then refer to $\log p(\mathbf{Y} \mid \boldsymbol{\theta})$ as the *incomplete data log likelihood*, and this we have assumed to be intractable. The EM algorithm then proposes an iterative procedure for maximising the incomplete data log likelihood by first defining the cost function

$$\mathcal{Q}\left(\boldsymbol{\theta}, \boldsymbol{\theta}_{old}\right) = \int_{\mathcal{Z}} p(\mathbf{Z} \mid \mathbf{Y}, \boldsymbol{\theta}_{old}) \log p(\mathbf{Y}, \mathbf{Z} \mid \boldsymbol{\theta}) \mathrm{d}\mathbf{Z}, \tag{5.2}$$

this is the expectation of the complete data log likelihood as a function of the parameter where the expectation is taken with respect to the conditional distribution of the latent variables for some fixed parameter $\boldsymbol{\theta}_{old}$ and the data $\mathbf{Y}$. A new parameter is then obtained by maximising (5.2) – this is referred to as the *M-step*. We then reevaluate the posterior conditional of the latent variables at this new parameter value and then retake the expectation in (5.2), and this is the *E-step*. This process is repeated until convergence of the likelihood function (5.1).

For our purposes, we shall be interested in MAP estimation, rather than maximum likelihood estimation, and we need to include the contribution from the prior, which is assumed to be independent of the latent variables, to the (5.2) leading the objective function for the MAP problem given by

$$\mathcal{Q}_{map}(\boldsymbol{\theta}; \boldsymbol{\theta}_{old}) = \int_{\mathcal{Z}} \log p(\mathbf{Y}, \mathbf{Z} \mid \boldsymbol{\theta}) p(\mathbf{Z} \mid \mathbf{Y}, \boldsymbol{\theta}_{old}) \mathrm{d}\mathbf{Z} + \log p(\boldsymbol{\theta}), \tag{5.3}$$

the general form of the EM-algorithm for MAP estimation is given by Algorithm 1.

For the rest of this chapter we shall make use of the angle-bracket notation for the expectation operator which for a random variable $\mathbf{Z}$ with density $q(\mathbf{Z})$, and a function

h depending on $\mathbf{Z}$, we write as

$$\langle h(\mathbf{Z}) \rangle_{q(\mathbf{Z})} \triangleq \int_{\mathcal{Z}} q(\mathbf{Z}) h(\mathbf{Z}) \mathrm{d}\mathbf{Z}.$$

```
Data: Observations $\mathbf{Y}$
Result: MAP Parameters $\hat{\boldsymbol{\theta}}$
while Parameters not converged do
    begin E-step
        Evaluate $p(\mathbf{Z} \mid \mathbf{Y}, \boldsymbol{\theta}_{old})$
    end
    begin M-step
        Evaluate $\boldsymbol{\theta}_{new}$ given by
            $\mathrm{s}ymbol\theta_{new} = \arg\max_{\boldsymbol{\theta}} \mathcal{Q}_{map}(\boldsymbol{\theta}; \boldsymbol{\theta}_{old})$
        where
            $\mathcal{Q}_{map}(\boldsymbol{\theta}; \boldsymbol{\theta}_{old}) = \langle \log p(\mathbf{Y}, \mathbf{Z} \mid \boldsymbol{\theta}) \rangle_{p(\mathbf{Z}|\mathbf{Y}, \boldsymbol{\theta}_{old})} + \log p(\boldsymbol{\theta})$.
    end
end
```

Algorithm 1: The EM algorithm for MAP estimation

5.2.1 EM algorithm for the MLFM-AG method

Having summarised the general EM algorithm, we now present the details for the specific implementation in the case of the adaptive gradient matching method considered in Chapter 3. We shall see that the optimisation problem is not significantly different to that given by optimising the marginal density presented in Section 3.4, but we include it here for completeness and in anticipation of the variational method developed in the next section.

For the MLFM-AG method, our parameter vector is taken to correspond to the latent forces and the connection coefficients $\{\mathbf{g}, \mathbf{B}\}$, and the nuisance parameters corresponding to the hyperparameters of the state interpolants and the constraint of the gradient expert $\{\boldsymbol{\phi}, \boldsymbol{\gamma}\}$. We will also consider the parameters of the noise distribution $\{\boldsymbol{\sigma}\}$, and for the construction of the MAP estimators we will need to consider, at the very least, the hyperparameters $\{\boldsymbol{\psi}\}$, of the latent forces. In practice we will usually want to assign hyperparameters on the connection coefficients $\mathbf{B}$ and so denote these hyperparameters by $\boldsymbol{\zeta}$. We, therefore, have the complete set of parameters for the adaptive gradient matching method

$$\boldsymbol{\theta} = \{\mathbf{g}, \mathbf{B}, \boldsymbol{\psi}, \boldsymbol{\phi}, \boldsymbol{\gamma}, \boldsymbol{\sigma}, \boldsymbol{\zeta}\}.$$

The set of latent variables which complete this model will be the trajectory variables $\mathbf{x} = (\mathbf{x}_1, \ldots, \mathbf{x}_K)^\top$, we chose to denote these variables by $\mathbf{X}$ rather than $\mathbf{Z}$ to remain consistent with the notation used in Chapter 3. We recall that the conditional distribution of the latent states was given by

$$p(\mathbf{x} \mid \mathbf{y}, \boldsymbol{\theta}) = \mathcal{N}(\mathbf{x} \mid \boldsymbol{\Sigma}[\operatorname{diag}(\boldsymbol{\tau}) \otimes \mathbf{I}]\mathbf{y}, \boldsymbol{\Sigma}), \tag{5.4}$$

where $\boldsymbol{\Sigma} = \boldsymbol{\Sigma}(\mathbf{g}, \mathbf{B}, \boldsymbol{\tau}, \gamma, \boldsymbol{\phi})$ is defined by (3.28), as in the previous chapters it will be more convenient to work with the vector of precisions, $\boldsymbol{\tau}$, rather than the vector of scale parameters, $\boldsymbol{\sigma}$. The density (5.4) determines the distribution with which the expectation is taken with respect to in the E-step of the EM algorithm as described above.

To construct the objective function for the M-step, we use the additive property of the error model to decompose the complete data log-likelihood term as

$$\begin{aligned}\log p(\mathbf{X}, \mathbf{Y} \mid \boldsymbol{\theta}) &= \log p(\mathbf{Y} \mid \mathbf{X}, \boldsymbol{\tau}) + \log p(\mathbf{X} \mid \mathbf{g}, \mathbf{B}, \boldsymbol{\phi}, \gamma), \\ &= \log \mathcal{N}(\mathbf{y} \mid \mathbf{x}, \operatorname{diag}(\boldsymbol{\tau}^{\circ -1}) \otimes \mathbf{I}_N) + \log \mathcal{N}(\mathbf{x} \mid \mathbf{0}, (\boldsymbol{\Lambda}_{ode} + \mathbf{C}_{\phi}^{-1})^{-1}), \end{aligned} \tag{5.5}$$

where $\boldsymbol{\Lambda}_{ode} = \boldsymbol{\Lambda}_{ode}(\boldsymbol{\theta})$ is given by (3.25), and $\mathbf{C}_{\phi}$ is the block diagonal inverse covariance matrix from the GP prior of the state interpolants. The complete data likelihood term (5.5) factors into the sum of two terms, one arising from the observation distribution and one from the ODE model and these two components are independent conditional on the trajectory variables as can be observed by the separation in the model factor graph, Figure 3.2a. It follows that we may write the total objective function for the MAP estimation problem as

$$Q_{map}(\boldsymbol{\theta}; \boldsymbol{\theta}_{old}) = Q_{obs}(\tau\ \boldsymbol{\theta}_{old}) + Q_{ode}(\mathbf{g}, \mathbf{B}, \boldsymbol{\phi}, \gamma; \boldsymbol{\theta}_{old}) + Q_{prior}(\mathbf{g}, \mathbf{B}, \boldsymbol{\psi}; \boldsymbol{\theta}_{old}), \tag{5.6}$$

where the components are given by

$$\mathcal{Q}_{obs}(\boldsymbol{\tau}, \boldsymbol{\theta}_{old}) = \langle \log p(\mathbf{Y} \mid \mathbf{X}, \boldsymbol{\tau}) \rangle_{p(\mathbf{X} \mid \mathbf{Y}, \boldsymbol{\theta}_{old})}. \tag{5.7a}$$

$$\mathcal{Q}_{ode}(\mathbf{g}, \mathbf{B}, \boldsymbol{\phi}, \gamma; \boldsymbol{\theta}_{old}) = \langle \log p(\mathbf{X} \mid \mathbf{g}, \mathbf{B}, \boldsymbol{\phi}, \gamma) \rangle_{p(\mathbf{X} \mid \mathbf{Y}, \boldsymbol{\theta}_{old})}, \tag{5.7b}$$

$$\mathcal{Q}_{prior}(\mathbf{g}, \mathbf{B}, \boldsymbol{\psi}, \boldsymbol{\zeta}; \boldsymbol{\theta}_{old}) = \log p(\mathbf{g} \mid \boldsymbol{\psi}) + \log p(\mathbf{B} \mid \boldsymbol{\zeta}). \tag{5.7c}$$

We now consider using the objective functions (5.7) to carry out the iterative EM scheme described above to estimate the model parameters. Because of the model's conditional independence structure, the vector of precision variable enters the objective function only through the observation noise model, and so to estimate these variables we need to consider only optimisation of the function

$$\begin{aligned}\mathcal{Q}_{obs}(\boldsymbol{\tau}; \boldsymbol{\theta}_{old}) &= \int_{\mathcal{X}} \log p(\mathbf{Y} \mid \mathbf{X} \mid \boldsymbol{\tau}) p(\mathbf{X} \mid \mathbf{Y}, \boldsymbol{\theta}_{old}) \mathrm{d}\mathbf{X} \\ &= \int_{\mathcal{X}_k} \prod_{k=1}^{K} \mathcal{N}(\mathbf{y}_k \mid \mathbf{x}_k, \tau_k^{-1}) p(\mathbf{x}_k \mid \mathbf{Y}, \boldsymbol{\theta}_{old}) \mathrm{d}\mathbf{x}_k \\ &= -\frac{1}{2} \left(\sum_{k=1}^{K} \tau_k \langle (\mathbf{y}_k - \mathbf{x}_k)^\top (\mathbf{y}_k - \mathbf{x}_k) \rangle_{p(\mathbf{X} \mid \mathbf{Y}, \boldsymbol{\theta}_{old})} - N \log \tau_k \right) + \text{const..}\end{aligned} \tag{5.8}$$

The optimisation of (5.8) leading to an update for the precision of the observations which will be given by

$$\tau_k^{-1} = N^{-1} \langle (\mathbf{y}_k - \mathbf{x}_k)^\top (\mathbf{y}_k - \mathbf{x}_k) \rangle_{p(\mathbf{x}_k | \mathbf{Y}, \boldsymbol{\theta}_{old})}. \tag{5.9}$$

It may also be desirable to include a prior on $\boldsymbol{\tau}$, for example, the gamma distribution was considered in Chapter 3, however, it is easy to do this by adding the contribution from the prior to the objective function (5.8) and we omit the details.

If we now consider the component of the full objective function (5.6) then the contribution, $\mathcal{Q}_{ode}$, arising the from the model specification is given by taking the expectation of the conditional log-likelihood of the trajectory variable , $p(\mathbf{x} \mid \mathbf{g}, \mathbf{B}, \boldsymbol{\phi}, \boldsymbol{\gamma})$ with respect to the posterior conditional distribution (3.24) evaluated at the old parameters and is given explicitly by

$$\begin{aligned}
\mathcal{Q}_{ode}(\mathbf{g}, \mathbf{B}, \boldsymbol{\phi}, \boldsymbol{\gamma}; \boldsymbol{\theta}_{old}) =& \langle \log p(\mathbf{X} \mid \mathbf{g}, \mathbf{B}, \boldsymbol{\phi}, \boldsymbol{\gamma}) \rangle_{p(\mathbf{X}|\mathbf{Y}, \boldsymbol{\theta}_{old})} \\
=& \langle \log \mathcal{N}(\mathbf{x} \mid \mathbf{0}, (\mathbf{C}_{\boldsymbol{\phi}} + \boldsymbol{\Lambda}_{ode}(\boldsymbol{\theta}))^{-1}) \rangle_{p(\mathbf{X}|\mathbf{Y}, \boldsymbol{\theta}_{old})} \\
=& -\frac{1}{2} \operatorname{Tr} \left(\langle \mathbf{x}\mathbf{x}^\top \rangle_{p(\mathbf{X}|\mathbf{Y}, \boldsymbol{\theta}_{old})} \left\{ \mathbf{C}_{\boldsymbol{\phi}}^{-1} + \sum_{k=1}^{K} \Lambda_k(\mathbf{g}, \mathbf{B}, \boldsymbol{\phi}_k, \gamma_k) \right\} \right) \\
&+ \frac{1}{2} \log \det \left(\mathbf{C}_{\boldsymbol{\phi}}^{-1} + \sum_{k=1}^{K} \Lambda_k(\mathbf{g}, \mathbf{B}, \boldsymbol{\phi}_k, \gamma_k) \right),
\end{aligned} \tag{5.10}$$

where each of the matrices $\boldsymbol{\Lambda}_k$ is given by (3.23). This combined with the contribution of the prior determines the objective function for the latent force variables, the connection coefficients and the remaining nuisance parameters.

Recalling from (3.22) that the entries of the matrix $\Lambda_{ode}(\mathbf{g}, \boldsymbol{\phi}, \boldsymbol{\gamma})$ are quadratic in the latent forces and connection coefficients then the first term of (5.10) is also a quadratic in these variables. Unfortunately, it is not possible to construct a closed form solution to the optimisation problem because of the relatively complex second term involving the determinant of the covariance matrix. While an explicit update equation is not available, it is still possible to calculate the gradient of the inverse covariance matrix with respect to the model parameters, and so use gradient-based optimisation methods. For the parameters, $\mathbf{g}$ and $\mathbf{B}$ the linear dependence of Λ_{ode} on these variables makes the gradient calculation relatively straightforward, although for datasets of larger dimensions this may become problematic.

In summary, there is relatively little to be gained by using the EM algorithm to estimate the MLFM-AG model, functionally (5.10) is very similar in form to the marginal likelihood (3.45). It follows that the optimisation of all the model parameters, apart from the observation precisions, must be done using gradient-based methods on an objective function that, in terms of computational complexity, is equivalent to maximising the marginalised conditional posterior presented in Section 3.4. Use of the EM algorithm, in this case, will provide at most a principled way to estimate the parameters governing the observation distribution separately from the remaining variables. In this instance the observation distribution is relatively simplistic involving only the K parameters $\tau_k, k = 1, \ldots, K$, however this observation would be more useful if, for example, we were to consider more complex observation models in which case the fact that the EM algorithm better respects the model conditional independence structure is likely to lead to an optimisation routine with better performance.

5.2.2 EM algorithm for the MLFM-SA method

While the EM algorithm for the MLFM-AG method was not significantly different to the process of direct maximisation of the marginal likelihood term we shall show there is a more significant benefit to be found using the EM algorithm to estimate the MLFM-SA method.

For the case of the MLFM-SA method our parameter vector will again include the latent forces and their hyperparameters, as well as the connection coefficients, however now rather than the hyperparameters of the GP state interpolants we must now consider the hyperparameters of the initial state approximation. For the development in this chapter we shall consider the case where the initial distribution is given by the constant model discussed in Section 4.3.3 which we recall has a degenerate distribution of the form

$$p(\mathbf{z}^{(0)} \mid \boldsymbol{\phi}) = \prod_{k=1}^{K} \mathcal{N}(\mathbf{z}^{(0)} \mid \mathbf{0}, \phi_k^{-1}\mathbf{1}_N\mathbf{1}_N^{\top}),$$

which constrains each dimension of the initial distribution to be constant valued. We also need to introduce the covariance matrix parameterising the noise of the transition distribution, $\boldsymbol{\Gamma}$, however in accordance with our discussion in 4.3.3 we prefer to regard this parameter as a regularisation parameter rather than as a model parameter. Because of this remark we will not look to optimise the transition covariance concurrently with the remaining parameters, and therefore for notational convenience we will suppress the dependence on $\boldsymbol{\Gamma}$ of the density functions considered in this section. Disregarding the transition covariance matrix our complete parameter set for this model is given by

$$\boldsymbol{\theta} = \{\mathbf{g}, \mathbf{B}, \boldsymbol{\psi}, \boldsymbol{\phi}, \boldsymbol{\sigma}, \boldsymbol{\zeta}\}. \tag{5.11}$$

To completely specify the MLFM-SA model we must also fix an initial approximation order, M, and an arbitrary fixed initial condition, τ_ν. We shall assume this is done beforehand and treat these variables as fixed during the optimisation process.

As was the case previously it shall be useful to consider the decomposition of the complete objective function into the components for different parameters. Just as for the adaptive gradient matching method we have the decomposition into the ODE model term and the observation noise term, but now the distribution of the model trajectory is given by the linear Gaussian dynamical system introduced in Section 4.3.3 which, owing to the Markov structure, admits the further decomposition

$$\begin{aligned}\log p(\mathbf{Z} \mid \boldsymbol{\theta}) &= \log p(\mathbf{z}^{(0)} \mid \boldsymbol{\theta}) + \sum_{m=1}^{M} \log p(\mathbf{z}^{(m)} \mid \mathbf{z}^{(m-1)}, \boldsymbol{\theta}) \\ &= \log p(\mathbf{z}^{(0)} \mid \boldsymbol{\phi}) + \sum_{m=1}^{M} \log p(\mathbf{z}^{(m)} \mid \mathbf{z}^{(m-1)}, \mathbf{g}, \mathbf{B}) \end{aligned} \tag{5.12}$$

thereby enabling us to separate the variables $\boldsymbol{\phi}$ from the additional model parameters and therefore leading to the full decomposition

$$\mathcal{Q}_{map}(\boldsymbol{\theta}; \boldsymbol{\theta}_{old}) = \mathcal{Q}_{obs}(\boldsymbol{\tau}) + \mathcal{Q}_{ode}(\mathbf{g}, \mathbf{B}, \boldsymbol{\Gamma}) + \mathcal{Q}_{init}(\boldsymbol{\phi}) + \mathcal{Q}_{prior}(\mathbf{g}, \mathbf{B}, \boldsymbol{\psi}, \boldsymbol{\zeta}) \tag{5.13}$$

where we have defined

$$\begin{aligned}
\mathcal{Q}_{obs} &= \langle \log p(\mathbf{Y} \mid \mathbf{Z}^{(M)}, \boldsymbol{\tau})\rangle_{p(\mathbf{Z}|\mathbf{Y},\boldsymbol{\theta}_{old})} && \text{(5.14a)}\\
\mathcal{Q}_{init}(\boldsymbol{\phi}, \boldsymbol{\theta}_{old}) &= \langle \log p(\mathbf{z}^{(0)} \mid \boldsymbol{\phi})\rangle_{p(\mathbf{Z}|\mathbf{Y},\boldsymbol{\theta}_{old})} && \text{(5.14b)}\\
\mathcal{Q}_{ode}(\mathbf{g}, \mathbf{B}; \boldsymbol{\theta}_{old}) &= \langle \log p(\mathbf{Z} \mid \mathbf{g}, \mathbf{B}, \rangle_{p(\mathbf{Z}|\mathbf{Y},\boldsymbol{\theta}_{old})} && \text{(5.14c)}\\
\mathcal{Q}_{prior}(\mathbf{g}, \mathbf{B}, \boldsymbol{\psi}, \boldsymbol{\zeta}) &= \log p(\mathbf{g} \mid \boldsymbol{\psi}) + \log p(\mathbf{B} \mid \zeta). && \text{(5.14d)}
\end{aligned}$$

Compared with the corresponding functions for the adaptive gradient model we note that the parameter of the initial Gaussian state approximation, $\boldsymbol{\phi}$, has been partitioned off to it's own function. This further isolation is possible because of the ideal conditional dependence structure allowed by the Markov chain structure of the trajectory distribution.

To provide a simpler closed form for (5.14c) we first consider an arbitrary density $q(\mathbf{z}^{(m-1)}, \mathbf{z}^{(m)})$ then for a single component $\log p(\mathbf{z}^{(m)} \mid \mathbf{z}^{(m-1)}, \boldsymbol{\theta})$ in (5.12) we can compute the following expectation

$$\begin{aligned}
\langle \log \mathcal{N}(\mathbf{z}^{(m)} \mid \mathbf{P}\mathbf{z}^{(m-1)})\rangle_{q(\mathbf{z}^{(m-1)},\mathbf{z}^{(m)})} &= -\frac{1}{2}\langle (\mathbf{z}_m - \mathbf{P}\mathbf{z}_{m-1})\boldsymbol{\Gamma}^{-1}(\mathbf{z}_m - \mathbf{P}\mathbf{z}_{m-1})\rangle_q(\mathbf{z}^{(m-1)}, \mathbf{z}_m)\\
&\quad - \frac{1}{2}\log\det\boldsymbol{\Gamma} + \text{const.}\\
&= -\frac{1}{2}\operatorname{Tr}(\langle \mathbf{z}^{(m-1)}\mathbf{z}^{(m-1)\top}\rangle_{q(\mathbf{z}_m)}\mathbf{P}^\top\boldsymbol{\Gamma}^{-1}\mathbf{P})\\
&\quad + \operatorname{Tr}(\langle \mathbf{z}^{(m-1)}\mathbf{z}^{(m)}\rangle_{q(\mathbf{z}^{(m-1)},\mathbf{z}^{(m)})}\boldsymbol{\Gamma}^{-1}\mathbf{P})\\
&\quad - \frac{1}{2}\log\det\boldsymbol{\Gamma} + \text{const.}, && \text{(5.15)}
\end{aligned}$$

where the constant term does not depend on $\mathbf{g}$ or $\mathbf{B}$, but does depend on $\boldsymbol{\Gamma}$. Depending on whether a representation is desired in terms of $\mathbf{g}$ or $\mathbf{B}$, can then be further rewritten by vectorising the operator $\mathbf{P}$ and then rearranging. To construct closed form expressions for the update equations of the model parameters it will be useful to again define the statistics

$$\boldsymbol{\Psi}_0 = \sum_{m=1} \mathbf{z}^{(i)}\mathbf{z}^{(i)\top}, \qquad \boldsymbol{\Psi}_1 = \sum_{m=1}^{M} \mathbf{z}^{(i-1)}\mathbf{z}^{(i)\top}, \tag{5.16}$$

then we may write the contribution to the objective function arising from the ODE model as

$$\begin{aligned}
\mathcal{Q}_{ode}(\mathbf{g},\mathbf{B}) &= \sum_{m=1}^{M} \langle \log \mathcal{N}(\mathbf{z}^{(m)} \mid \mathbf{P}\mathbf{z}^{(m-1)}, \boldsymbol{\Gamma}) \rangle_{p(\mathbf{Z}|\mathbf{Y},\boldsymbol{\theta}_{old})} \\
&= -\frac{1}{2}\operatorname{Tr}\left(\langle \boldsymbol{\Psi}_0 \rangle_{p(\mathbf{Z}|\mathbf{Y},\boldsymbol{\theta}_{old})} \mathbf{P}^\top \boldsymbol{\Gamma}^{-1}\mathbf{P}\right) \\
&\quad + \operatorname{Tr}\left(\langle \boldsymbol{\Psi}_1 \rangle_{p(\mathbf{Z}|\mathbf{Y},\boldsymbol{\theta}_{old})} \boldsymbol{\Gamma}^{-1}\mathbf{P}\right) \\
&\quad - \frac{1}{2}\log\det\boldsymbol{\Gamma} + \text{const.} \\
&= -\frac{1}{2}\operatorname{vec}(\mathbf{P})^\top \left(\langle \boldsymbol{\Psi}_0 \rangle_{p(\mathbf{Z}|\mathbf{Y},\boldsymbol{\theta}_{old})} \otimes \boldsymbol{\Gamma}^{-1}\right)\operatorname{vec}(\mathbf{P}) \\
&\quad + \operatorname{vec}(\boldsymbol{\Gamma}^{-1}\langle \boldsymbol{\Psi}_1^\top \rangle_{p(\mathbf{Z}|\mathbf{Y},\boldsymbol{\theta}_{old})})^\top \operatorname{vec}(\mathbf{P}) \\
&\quad - \frac{1}{2}\log\det\boldsymbol{\Gamma} + \text{const.}
\end{aligned} \tag{5.17}$$

The expression (5.17) is quadratic in the entries of the vectorisation of $\mathbf{P}$, and therefore after conditioning on $\mathbf{B}$, respectively $\mathbf{g}$, we will obtain a quadratic in $\mathbf{g}$, respectively $\mathbf{B}$. For the update of the latent force variableswe use the representation $\mathbf{P} = \mathbf{V}\mathbf{g} + \mathbf{v}_0$, where $\mathbf{V}$ and $\mathbf{v}_0$ are defined by (4.21). Then using the expression (5.17) along with the prior on $\mathbf{g}$ we can construct the quadratic

$$\begin{aligned}
Q_{ode} + \ln p(\mathbf{g} \mid \boldsymbol{\psi}) &= -\frac{1}{2}\operatorname{vec}(\mathbf{P})^\top (\langle \boldsymbol{\Psi}_0 \rangle_{p(\mathbf{Z}|\mathbf{Y},\boldsymbol{\theta}_{old})} \otimes \boldsymbol{\Gamma}^{-1})\operatorname{vec}(\mathbf{P}) - \frac{1}{2}\mathbf{g}^\top \mathbf{C}_{\boldsymbol{\psi}}^{-1}\mathbf{g} + \text{const.} \\
&\quad + \operatorname{vec}(\boldsymbol{\Gamma}^{-1}\langle \boldsymbol{\Phi}_1 \rangle_{p(\mathbf{Z}|\mathbf{Y},\boldsymbol{\theta}_{old})})^\top \operatorname{vec}(\mathbf{P}) \\
&= -\frac{1}{2}\mathbf{g}^\top \left(\mathbf{V}^\top (\langle \boldsymbol{\Psi}_0 \rangle_{p(\mathbf{Z}|\mathbf{Y},\boldsymbol{\theta}_{old})} \otimes \boldsymbol{\Gamma}^{-1})\mathbf{V} + \mathbf{C}_{\boldsymbol{\psi}}^{-1}\right)\mathbf{g} \\
&\quad + \Bigg[\operatorname{vec}(\boldsymbol{\Gamma}^{-1}\langle \boldsymbol{\Psi}_1^\top \rangle_{p(\mathbf{Z}|\mathbf{Y},\boldsymbol{\theta}_{old})}) \\
&\quad - \mathbf{v}_0^\top (\langle \boldsymbol{\Psi}_0 \rangle_{p(\mathbf{Z}|\mathbf{Y},\boldsymbol{\theta}_{old})}) \otimes \boldsymbol{\Gamma}^{-1}) \Bigg] \mathbf{V}\mathbf{g} + \text{const.},
\end{aligned} \tag{5.18}$$

and therefore, for given values of $\mathbf{B}$ and $\boldsymbol{\Gamma}$, the optimum, $\hat{\mathbf{g}}$, of (5.18) is given by the solution to the linear system of equations

$$\begin{aligned}
\left[\mathbf{V}^\top (\langle \boldsymbol{\Psi}_0 \rangle_{p(\mathbf{Z}|\mathbf{Y},\boldsymbol{\theta}_{old})} \otimes \boldsymbol{\Gamma}^{-1})\mathbf{V} + \mathbf{C}_{\boldsymbol{\psi}}^{-1}\right] \hat{\mathbf{g}} = \Bigg[&\operatorname{vec}(\boldsymbol{\Gamma}^{-1}\langle \boldsymbol{\Psi}_1^\top \rangle_{p(\mathbf{Z}|\mathbf{Y},\boldsymbol{\theta}_{old})}) \\
&- \mathbf{v}_0^\top (\langle \boldsymbol{\Psi}_0 \rangle_{p(\mathbf{Z}|\mathbf{Y},\boldsymbol{\theta}_{old})}) \otimes \boldsymbol{\Gamma}^{-1}) \Bigg] \mathbf{V}.
\end{aligned} \tag{5.19}$$

Similarly if we assume that the prior on $\mathbf{B}$ is given by a mean zero Gaussian distribution $p(\mathbf{B} \mid \boldsymbol{\zeta})$ with covariance matrix $\mathbf{C}_{\boldsymbol{\zeta}}$ then the optimal value, $\hat{\mathbf{b}}$, now for known values of the latent forces and transition covariance matrix, is given by

$$\begin{aligned}
\left[\mathbf{W}^\top \left(\langle \boldsymbol{\Phi}_0 \rangle_{p(\mathbf{Z}|\mathbf{Y},\boldsymbol{\theta}_{old})} \otimes \boldsymbol{\Gamma}^{-1}\right)\mathbf{W} + \mathbf{C}_{\zeta}^{-1}\right] \hat{\mathbf{b}} = \Bigg[&\operatorname{vec}(\boldsymbol{\Gamma}^{-1}\langle \boldsymbol{\Phi}_1 \rangle_{p(\mathbf{Z}|\mathbf{Y},\boldsymbol{\theta}_{old})}) \\
&- \mathbf{w}_0^\top \left(\langle \boldsymbol{\Phi}_0 \rangle_{p(\mathbf{Z}|\mathbf{Y},\boldsymbol{\theta}_{old})} \otimes \boldsymbol{\Gamma}^{-1}\right) \Bigg] \mathbf{W}.
\end{aligned} \tag{5.20}$$

We can also derive closed form estimates for the transition covariance matrix following the same process, however as discussed in Section 4.3.3 these parameters were introduced primarily as regularisation parameters, much like the parameters $\boldsymbol{\gamma}$ in the adaptive gradient matching method, and as such these variables are arguably better held fixed and the dependence of the resulting inference for the remaining parameters determined by doing a sensitivity analysis after model fitting.

Finally we also need to consider the initial contributions to the model

$$\begin{aligned}\mathcal{Q}_{init}(\boldsymbol{\phi}; \boldsymbol{\theta}_{old}) &= \sum_{k=1}^{K} \langle \log \mathcal{N}(\mathbf{z}_k^{(0)} \mid \mathbf{0}, \phi_k \mathbf{1}\mathbf{1}^\top) \rangle_{p(\mathbf{z}_k^{(0)} | \mathbf{Y}, \boldsymbol{\theta}_{old})} + \log p(\boldsymbol{\phi}) \\ &= -\frac{1}{2} \sum_{k=1}^{K} \phi_k \langle (z_{k\nu}^{(0)})^2 \rangle_{p(\mathbf{z}_k^{(0)} | \mathbf{Y}, \boldsymbol{\theta}_{old})} + \log p(\boldsymbol{\phi}). \end{aligned} \tag{5.21}$$

Note the connection with (5.21) and the singular problem of estimating the MLE of the standard deviation of a Gaussian random variable from a single data point, therefore the solution of this problem is dependent on the prior, and poorly informed by the latent trajectory and therefore it is more appropriate to also consider this variable as an additional regularising hyperparameter.

```
Data: Observations $\mathbf{Y}$
Result: MAP Parameters $\hat{\boldsymbol{\theta}} = \{\hat{\mathbf{g}}, \hat{\mathbf{B}}, \hat{\boldsymbol{\phi}}\}$
while Parameters not converged do
    begin E-Step
        for $m = 1, \ldots, M$ do
            Calculate
                $\langle \mathbf{z}^{(m-1)} \mathbf{z}^{(m-1)\top} \rangle_{p(\mathbf{z}^m | \mathbf{Y}, \boldsymbol{\theta}_{old})}$,    $\langle \mathbf{z}^{(m-1)} \mathbf{z}^{(m)\top} \rangle_{p(\mathbf{z}^m | \mathbf{Y}, \boldsymbol{\theta}_{old})}$,
             using the Kalman filter smoothing updates.
        end
    end
    begin M-Step
        $\hat{\mathbf{g}} \leftarrow g_{new}$ where $g_{new}$ solves (5.19);
        $\hat{\boldsymbol{\beta}} \leftarrow \boldsymbol{\beta}_{new}$ where $\beta_{new}$ solves (5.20);
    end
end
```

Algorithm 2: EM algorithm for MAP estimation in the MLFM-SA model using coordinate updates for the M-step

If we use a coordinate descent method that respects the conditional independence structure we are able to construct an optimisation procedure for the MLFM-SA model that is completely gradient free. Not only does this achieve our goal of avoiding expensive calculations of the gradient of the covariance matrix which was polynomial in the latent forces, but the nuisance parameters $\boldsymbol{\phi}$ and γ that have been introduced to the model to motivate our approximation also possess closed form updates. This is in contrast to the situation observed for the MLFM-AG model in which, using either the direct maximisation of the marginal likelihood or the EM approach, there were no immediately useful gradient based optimisation methods. While this feature is attractive it does not therefore follow that the MLFM-SA method will be numerically more efficient to optimise than the MLFM-AG method. While the linear dynamic system structure allows for an E-step that is computationally efficient it will still be relatively expensive for higher orders of approximation, but a more serious concern is the local character of the MLFM-SA method which automatically leads to a discounting in the likelihood function of points away from the fixed time. In principal it should be possible to address this issue using the mixture model introduced in Section 4.5 by adapting the approach in section, but we do not provide the details here.

5.3 Mean field variational inference

As discussed in the introduction to this chapter we shall be interested in deriving estimates of the distribution of the parameters in our MLFM. The intractable nature of the marginal likelihood terms prevents us from finding a simple analytic form for the conditional distributions, and so a natural alternative is to use Monte Carlo methods instead to obtain samples from these distributions. While such approaches are appealing in that, they provide guarantees on when we will be sampling from the correct distribution they may not be practical, and there are situations in which we may prefer faster deterministic approaches.

A particularly important class of deterministic approximations are the variational Bayesian methods [Bishop, 2006] which seek to replace the intractable distribution with an approximation chosen by optimising a function of the actual distribution. An instance in which this optimisation problem may be solved explicitly is the *mean field* approximation. The general setup involves the determination of an approximation to a distribution with conditional density function $p(\mathbf{Z}|\mathbf{Y})$ where $\mathbf{Z}$ is an arbitrary vector-valued random variable, and $\mathbf{Y}$ is a vector of observed values. In this instance, we are making no distinction between the model parameters and the latent variables unlike in the development of the EM algorithm. The mean field approximation attempts to find a representation of the distribution of $\mathbf{Z}$ by constructing a product distribution over a partition of this variable

$$p(\mathbf{Z}|\mathbf{Y}) \approx q(\mathbf{Z}) \triangleq \prod_{j=1}^{N} q_j(\mathbf{Z}_j), \tag{5.22}$$

where $\mathbf{Z}$ has been partitioned as $\mathbf{Z} = \{\mathbf{Z}_1, \ldots, \mathbf{Z}_N\}$. To construct an approximation that is in some sense optimal we need to introduce some suitable cost function, an important choice is a Kulback-Leiber divergence, which for two distributions with density $q(\cdot)$ and $p(\cdot)$ respectively is defined as [Kullback, 1968]

$$D_{KL}(q\|p) = -\int q(\mathbf{z}) \log \frac{q(\mathbf{z})}{p(\mathbf{z}|\mathbf{y})} \mathrm{d}\mathbf{Z}. \tag{5.23}$$

The goal is to find the optimal distribution $q^*(\mathbf{Z})$ that minimises this cost function subject to the given product structure — often the choice of the partition of the variable $\mathbf{Z}$ will be made in such a way as to make deriving the optimal choice feasible. It is easily seen using the calculus of variations [Jost and Li-Jost, 1998] that the optimal choice is given by $q^*(\mathbf{Z}) = \prod_{j=1}^{M} q(\mathbf{Z}_j^*)$, where each of the optimal factors are determined by

$$\log q_j^*(\mathbf{Z}_j) = \langle \log p(\mathbf{z}_j, \mathbf{z}_{-j}, \mathbf{y}) \rangle_{q_{-j}^*(\mathbf{Z}_{-j})}, \tag{5.24}$$

where $\mathbf{z}_j$ denotes the vector $\mathbf{z}$ with $\mathbf{z}_j$ removed and the expectation is taken with respect to the distribution

$$q_{-j}^*(\mathbf{z}_{-j}) = \prod_{i=1,\, i\neq j}^{M} q_i^*(\mathbf{z}_i).$$

The ease with which this method may be applied depends on the tractability of the expectation in (5.24) and furthermore how easy it is to normalise the resulting expression. This, in turn, is often dependent on the conditional independence structure of the model; indeed the method is closely related to Gibbs sampling. It is often the case that if it is straightforward to carry out Gibbs sampling it is also easy to construct the mean field variational approximation the same factorisation and this is the direction we shall pursue now where we shall show that for the latent states, $\mathbf{X}$, latent forces $\mathbf{g}$ and noise parameters $\boldsymbol{\sigma}$ it is straight forward to construct the approximation.

$$p(\mathbf{x}, \mathbf{g}, \boldsymbol{\sigma} | \boldsymbol{\gamma}, \boldsymbol{\phi}) \approx q_x(\mathbf{x}) q_g(\mathbf{g}) q_\sigma(\boldsymbol{\sigma}), \tag{5.25}$$

however as we remarked for the Gibbs sampling algorithm, there is no straight forward variational distribution for the state Gaussian process hyper-parameters $\boldsymbol{\phi}$ and the temperature parameter of the gradient expert $\boldsymbol{\gamma}$.

5.3.1 Mean field approximation of the MLFM-AG method

While the EM algorithm contributed relatively little to the problem of constructing point estimates for the MLFM-AG method, variational methods will prove much more useful for the problem of constructing distributional estimates. Indeed we shall see that because the optimal factors in mean field gradient matching method are obtained by integrating over the, logarithm of the, joint density $p(\mathbf{Y}, \mathbf{X}, \mathbf{g}, \mathbf{B})$, rather than the conditional density $p(\mathbf{X} \mid \mathbf{Y}, \mathbf{g}, \mathbf{B})$ used for the EM method, we will be able to obtain closed-form expressions. Our strategy will be to make use of the equivalent linear representations of the evolution equation *before* they get absorbed into the construction of the model (inverse) covariance matrix, as was done successfully for the calculation of the conditional distributions in Chapter 3.

We will construct the mean field approximation in a way that respects the natural conditional independence structure of the model as represented in Figure 3.2 which allowed for the tractable conditional distributions presented in Chapter 3. This implies an approximation to the distribution with a density which factors as

$$q(\mathbf{X}, \mathbf{g}, \mathbf{B}, \boldsymbol{\tau}) = q_x(\mathbf{X}) q_g(\mathbf{g}) q_\beta(\mathbf{B}) q_\tau(\boldsymbol{\tau}), \tag{5.26}$$

where the factors are defined jointly by

$$q_x(\mathbf{X}) \propto \exp\left\{ \langle \log p(\mathbf{X}, \mathbf{g}, \mathbf{B}, \mathbf{Y}, \boldsymbol{\tau}) \rangle_{q_g(\mathbf{g}) q_B(\mathbf{B}) q_\tau(\boldsymbol{\tau})} \right\}, \tag{5.27a}$$

$$q_g(\mathbf{g}) \propto \exp\left\{ \langle \log p(\mathbf{X}, \mathbf{g}, \mathbf{B}, \mathbf{Y}, \boldsymbol{\tau}) \rangle_{q_x(\mathbf{X}) q_B(\mathbf{B}) q_\tau(\boldsymbol{\tau})} \right\}, \tag{5.27b}$$

$$q_B(\mathbf{B}) \propto \exp\left\{ \langle \log p(\mathbf{X}, \mathbf{g}, \mathbf{B}, \mathbf{Y}, \boldsymbol{\tau}) \rangle_{q_x(\mathbf{X}) q_g(\mathbf{g}) q_\tau(\boldsymbol{\tau})} \right\}, \tag{5.27c}$$

$$q_\tau(\boldsymbol{\tau}) \propto \exp\left\{ \langle \log p(\mathbf{X}, \mathbf{g}, \mathbf{B}, \mathbf{Y}, \boldsymbol{\tau}) \rangle_{q_x(\mathbf{X}) q_g(\mathbf{g}) q_\beta(\mathbf{B})} \right\}. \tag{5.27d}$$

Given that we were unable to derive tractable conditional distributions for the hyperparameters of the state interpolating GPs or the latent force GPs it is unsurprising that these variables do not admit a tractable variational approximation either, and the same is true of the gradient expert parameter. As such we shall only consider the construction of an approximation to the distribution

$$q(\mathbf{X}, \mathbf{g}, \mathbf{B}, \boldsymbol{\tau}) \approx p(\mathbf{X}, \mathbf{g}, \mathbf{B}, \boldsymbol{\tau} \mid \mathbf{Y}, \boldsymbol{\phi}, \boldsymbol{\gamma}, \boldsymbol{\psi}),$$

with $\{\boldsymbol{\phi}, \boldsymbol{\gamma}, \boldsymbol{\psi}\}$ held fixed at some constant value – a natural choice would be the MAP estimates determined in Section 5.2.1. Alternatively they could be chosen using some appropriate model selection method.

Since the construction of the approximating distributions will follow a process similar to undertaken in Chapter 3 to construct the conditional distribution of the Gibbs sampling process we again require the various decomposition's of the flow function

$$\mathbf{f}_k = \sum_{j=1}^{K} \mathbf{u}_{kj} \circ \mathbf{x}_j = \sum_{r=1}^{R} \mathbf{v}_{kr} \circ \mathbf{g}_r = \sum_{r=1}^{R} \sum_{d=1}^{D} \mathbf{w}_{krd} \beta_{rd},$$

which were defined in (3.20). These representations all involve products of the three variables in which we are interested, but the independence assumption of the factorised distribution ensures that it will be each to calculate the mean and covariance of the vectors, $\mathbf{f}_k$, with respect to the approximate densities (5.27). We now present closed form expressions for each factor the variational Bayes approximation using the mean field approximation.

Mean field update of the latent states' distribution

Using the, by now, familiar decomposition of the joint density into the observation model and the ODE model we have

$$\begin{aligned}
\log q_x(\mathbf{X}) &= \langle \log p(\mathbf{Y}, \mathbf{X}, \mathbf{g}, \mathbf{B}, \boldsymbol{\tau} \mid \boldsymbol{\psi}, \boldsymbol{\gamma} \rangle + \text{const.} \\
&= \langle \log p(\mathbf{Y} \mid \mathbf{X}, \boldsymbol{\tau}) \rangle_{q_\tau(\boldsymbol{\tau})} + \langle \log p(\mathbf{X}, \mathbf{g}, \mathbf{B} \mid \boldsymbol{\psi}, \boldsymbol{\gamma}) \rangle_{q_g(\mathbf{g}) q_B(\mathbf{B})} \\
&= -\frac{1}{2} \operatorname{diag}(\langle \boldsymbol{\tau} \rangle) \otimes \mathbf{I}_N - \frac{1}{2} \sum_{k=1}^{K} \Big(\langle (\mathbf{f}_k - \mathbf{m}_k)^\top \mathbf{S}_k^{-1} (\mathbf{f}_k - \mathbf{m}_k) \rangle_{q(\mathbf{g}) q(\mathbf{B})} + \mathbf{x}_k^\top \mathbf{C}_{\phi_k}^{-1} \mathbf{x}_k \Big) + \text{const.} \\
&= -\frac{1}{2} \operatorname{diag}(\langle \boldsymbol{\tau} \rangle) \otimes \mathbf{I}_N - \frac{1}{2} \mathbf{x}^\top \left(\sum_{k=1}^{K} \langle \Lambda_k \rangle_{q_g(\mathbf{g}) q_B(\mathbf{B})} + \mathbf{C}_{\boldsymbol{\phi}}^{-1} \right) \mathbf{x} + \text{const.}, \qquad (5.28)
\end{aligned}$$

where $\boldsymbol{\Lambda_k}$ was defined by (3.23). To calculate the expected value of the quadratic form in (3.23) it will be useful to first define the $N \times N$ matrices

$$\begin{aligned}
\mathcal{U}_{kij} &\triangleq \langle \mathbf{u}_{ki} \mathbf{u}_{kj}^\top \rangle_{q_g(\mathbf{g}) g_B(\mathbf{B})} \\
&= \sum_{r,r'=1}^{R} \langle A_{rki} A_{r'kj} \mathbf{g}_r \mathbf{g}_s^\top \rangle_{q_g(\mathbf{g}) q_B(\mathbf{B})} \\
&= \sum_{r,r'=1}^{R} \sum_{d,d'} \langle \beta_{rd} \beta_{r'd'} \rangle_{q_B(\mathbf{B})} L_{dki} L_{d'kj} \langle \mathbf{g}_r \mathbf{g}_s^\top \rangle_{q_g(\mathbf{g})}, \qquad (5.29)
\end{aligned}$$

for $i, j = 1, \ldots, K$. As previously we have defined $\mathbf{g}_0$ to be a constant vector with all entries equal to unity, and so independent of the other variables. Then using (3.23) we have

$$q_x(\mathbf{X}) \propto \mathcal{N}(\mathbf{y} \mid \mathbf{x}, \mathrm{diag}(\langle \boldsymbol{\tau} \rangle^{\circ -1}_{q_\tau(\boldsymbol{\tau})}) \otimes \mathbf{I}_N) \times \mathcal{N}\left(\mathbf{x} \mid \mathbf{0}, \left(\mathbf{C}_\phi^{-1} + \boldsymbol{\Lambda}_{ode}^{VB}\right)^{-1}\right), \tag{5.30}$$

where $\boldsymbol{\Lambda}_{ode}^{VB}$ is the mean field variational Bayes equivalent of the matrix (3.25). That is to say

$$\boldsymbol{\Lambda}_{ode}^{VB} = \sum_{k=1}^{K} \boldsymbol{\Lambda}_{ode,k}^{VB},$$

where each $\boldsymbol{\Lambda}_{ode,k}^{VB}$ is $K \times K$ block matrix, and each block is the $N \times N$ matrix given by

$$\begin{aligned}\langle \boldsymbol{\Lambda}_{kij} \rangle_{q(\mathbf{g},\mathbf{B})} = &\,\mathcal{U}_{kij} \circ \mathbf{S}_{\phi_k}^{-1} \\ &- \delta_{ki} \,\mathrm{diag}(\langle \mathbf{u}_{ki} \rangle_{q(\mathbf{g})q(\mathbf{B})}) \mathbf{S}_{\phi_k}^{-1} \mathbf{M}_k \\ &- \delta_{kj} \mathbf{M}_k^\top \mathbf{S}_{\phi_k}^{-1} \,\mathrm{diag}(\langle \mathbf{u}_{kj} \rangle_{q(\mathbf{g})q(\mathbf{B})}) \\ &+ \mathbf{M}_k^\top \mathbf{S}_{\phi_k}^{-1} \mathbf{M}_k, \end{aligned} \tag{5.31}$$

for $i, j = 1, \ldots, K$. Normalising the product (5.30) we can conclude the optimal variational mean field factor for the state variable is given by

$$q^*(\mathbf{x}) = \mathcal{N}(\mathbf{x} \mid \mathbf{m}_x^{VB}, \mathbf{K}_x^{VB}), \tag{5.32}$$

where the parameters of this Gaussian distribution are given by

$$\mathbf{K}_x^{VB} = \left(\mathrm{diag}(\langle \boldsymbol{\tau} \rangle_{q^*(\boldsymbol{\tau})}) \otimes \mathbf{I}_N + \langle \Lambda_{ode} \rangle_{q^*(\mathbf{g})q^*(\mathbf{B})} + \mathbf{C}_{\boldsymbol{\phi}}^{-1}\right)^{-1}, \tag{5.33a}$$

$$\mathbf{m}_x^{VB} = \mathbf{K}_x^{VB} \left[\mathrm{diag}(\langle \boldsymbol{\tau} \rangle_{q(\boldsymbol{\tau})}) \otimes \mathbf{I}_N\right] \mathbf{y}. \tag{5.33b}$$

This is the variational equivalent of the Gibbs conditional distributions introduced in Chapter 3.

Mean field update of the latent forces' distribution

The mean field distribution of the latent force variables, $\mathbf{g}$, is calculated in a manner similar to that of the latent states, but unlike in the case above we cannot rearrange the general quadratic form arising from the ODE model into a homogeneous quadratic form. Instead we have an additional term arising from the offset matrix, $\mathbf{A}_0$, in the specification of the MLFM. As in the update for the latent states we will need to calculate the expectation of the quadratic form $(\mathbf{f}_k - \mathbf{m}_k)\mathbf{S}_k^{-1}(\mathbf{f}_k - \mathbf{m}_k)$, and so with this in mind it will be useful to now define the collection of $N \times N$ matrices

$$\begin{aligned}\mathcal{V}_{kr}^{(0)} \triangleq &\langle (\mathbf{m}_k - \mathbf{v}_{k0}) \mathbf{v}_{kr}^\top \rangle_{q_x(\mathbf{X})g_B(\mathbf{B})} \\ = &\mathbf{M}_k \sum_{i=1}^{K} \langle \mathbf{x}_k \mathbf{x}_i^\top \rangle_{q(\mathbf{X})} \sum_{d=1}^{D} \langle \beta_{rd} L_{dki} \rangle_{q(\mathbf{B})} \\ &- \sum_{i,j=1}^{K} \sum_{d,d'=1}^{D} L_{dki} L_{dkj} \langle \beta_{rd} \beta_{rd'} \rangle_{q(\mathbf{B})} \langle \mathbf{x}_i \mathbf{x}_j^\top \rangle_{q(\mathbf{X})}, \end{aligned} \tag{5.34}$$

where $\mathbf{m}_k = \mathbf{M}_k \mathbf{x}_k$ was defined by (3.6a), and

$$\begin{aligned}\mathcal{V}^{(1)}_{krr'} &\triangleq \langle \mathbf{v}_{kr}\mathbf{v}_{kr'}^{\top}\rangle_{q(\mathbf{x})q(\mathbf{B})} \\ &= \sum_{i,j=1}^{K}\sum_{d,d'=1}^{K} L_{dki}L_{dkj}\langle \beta_{rd}\beta_{rd'}\rangle_{q(\mathbf{B})}\langle \mathbf{x}_i\mathbf{x}_j^{\top}\rangle_{q(\mathbf{x})}.\end{aligned} \tag{5.35}$$

Then on taking the expectation we have

$$\begin{aligned}\langle(\mathbf{f}_k - \mathbf{m}_k)^{\top}\mathbf{S}_k^{-1}(\mathbf{f}_k - \mathbf{m}_k)\rangle_{q(\mathbf{x})q(\mathbf{B})} =& \sum_{r=1}^{R}\sum_{r'=1}^{R} \mathbf{g}_r^{\top}\mathcal{V}^{(1)}_{krr'} \circ \mathbf{S}^{-1}_{\phi_k,\gamma_k}\mathbf{g}_r' \\ &- 2\sum_{r=1}^{R}\mathcal{V}^{(0)}_{kr} \circ \mathbf{S}^{-1}_{\phi_k,\gamma_k}\mathbf{1}_N \end{aligned}\tag{5.36}$$

Combining these results, and including the contribution from the prior we have that

$$\begin{aligned}\log q(\mathbf{g}) =& \langle \log p(\mathbf{x} \mid \mathbf{g}, \mathbf{B})\rangle_{q(\mathbf{x})q(\mathbf{B})} + \log p(\mathbf{g}) + \text{const.} \\ =& -\frac{1}{2}\sum_{k=1}^{K}\Big\langle (\mathbf{f}_k - \mathbf{m}_k)^{\top}\mathbf{S}^{-1}_{\phi_k}(\mathbf{f}_k - \mathbf{m}_k)\Big\rangle + \text{const.} \\ =& -\frac{1}{2}\sum_{r,r'=1}^{R}\mathbf{g}_r^{\top}\left(\sum_{k=1}^{K}\mathcal{V}^{(1)}_{krr'} \circ \mathbf{S}^{-1}_{\phi_k} + \delta_{rr'}\mathbf{C}^{-1}_{\psi_r}\right)\mathbf{g}_{r'} \qquad (5.37) \\ &+ \sum_{r=1}^{R}\mathbf{g}_r^{\top}\left(\sum_{k=1}^{K}\mathcal{V}^{(0)}_{kr} \circ \mathbf{S}_k^{-1}\right)\mathbf{1}_N + \text{const.} \qquad (5.38)\end{aligned}$$

This implies that the optimal log-likelihood for the latent forces in the mean field variational Bayes approximations to the MLFM-AG model will be a quadratic, and therefore we can conclude

$$q^{*}(\mathbf{g}) = \mathcal{N}(\mathbf{g} \mid \mathbf{m}_g^{VB}, \mathbf{K}_g^{VB}) \tag{5.39}$$

The covariance matrix, $\mathbf{K}_g^{VB}$, is constructed by inverting the the $R \times R$ matrix with blocks

$$\sum_{k=1}^{K}\mathcal{V}^{(1)}_{krr'} \circ \mathbf{S}_k^{-1} + \delta_{rr'}\mathbf{C}^{-1}_{\psi_r}, \tag{5.40}$$

and the mean is given by

$$\mathbf{m}_g^{VB} = \mathbf{K}_g^{VB}\sum_{k=1}^{K}\begin{bmatrix}\mathcal{V}^{(0)}_{k1} \circ \mathbf{S}_k^{-1}\mathbf{1}_N \\ \vdots \\ \mathcal{V}^{(0)}_{kR} \circ \mathbf{S}_k^{-1}\mathbf{1}_N\end{bmatrix}. \tag{5.41}$$

Mean field distribution for the connection coefficients

The construction of the variational distribution of the connection coefficients proceeds in a similar way to that already considered for the latent state, and latent forces. We first define the $N \times N$ matrices

$$\begin{aligned}\mathcal{W}^{(0)}_{krd} &\triangleq \langle \mathbf{m}_k \mathbf{w}_{krd}^\top \rangle_{q(\mathbf{X})q(\mathbf{g})} \\ &= \mathbf{M}_k \sum_{j=1}^{K} L_{djk} \langle \mathbf{x}_k \mathbf{x}_j^\top \rangle_{q(\mathbf{X})} \operatorname{diag}(\langle \mathbf{g}_r \rangle_{q(\mathbf{g})}),\end{aligned}$$

and

$$\begin{aligned}\mathcal{W}^{(1)}_{krdr'd'} &\triangleq \langle \mathbf{w}_{kr'd'} \mathbf{w}_{krd}^\top \rangle_{q(\mathbf{g})q(\mathbf{X})} \\ &= \langle \mathbf{g}_{r'} \mathbf{g}_r^\top \rangle_{q(\mathbf{g})} \circ \sum_{i,j=1}^{K} L_{dki} L_{d'kj} \langle \mathbf{x}_i \mathbf{x}_j^\top \rangle_{q(\mathbf{X})},\end{aligned}$$

for $r, r' = 0, 1 \ldots, R$ and $d, d' = 1, \ldots, D$. We then have

$$\begin{aligned}\log q(\mathbf{B}) =& \langle \log p(\mathbf{X} \mid \mathbf{g}, \mathbf{B}) \rangle_{q(\mathbf{X})q(\mathbf{B})} + \log p(\mathbf{B} \mid \boldsymbol{\zeta}) \\ =& -\frac{1}{2} \langle (\mathbf{f}_k - \mathbf{m}_k)^\top \mathbf{S}_{\phi_k}^{-1} (\mathbf{f}_k - \mathbf{m}_k)^\top \rangle_{q(\mathbf{X})q(\mathbf{g})} - \frac{1}{2} \boldsymbol{\beta}^\top \mathbf{C}_\zeta^{-1} \boldsymbol{\beta} + \text{const.} \qquad (5.42) \\ =& -\frac{1}{2} \sum_{r,r'=0}^{R} \sum_{d,d'=1}^{D} \beta_{rd} \beta_{r'd'} \sum_{k=1}^{K} \operatorname{Tr}\left(\mathcal{W}^{(1)}_{krdr'd'} \mathbf{S}_{\phi_k}^{-1} \right) \\ &+ \sum_{r=0}^{R} \sum_{d=1}^{D} \beta_{rd} \sum_{k=1}^{K} \operatorname{Tr}\left(\mathcal{W}^{(0)}_{krd} \mathbf{S}_{\phi_k}^{-1} \right) - \frac{1}{2} \boldsymbol{\beta}^\top \mathbf{C}_\zeta^{-1} \boldsymbol{\beta} + \text{const.}, \qquad (5.43)\end{aligned}$$

which is once more a quadratic, this time in the variables β_{rd}.

Mean field distribution for the observation precisions

Finally we consider the variational approximation to the parameters of the observation error scale. As in Section 3.3.1 we shall place a Gamma prior with shape parameter, a_{k0}, and rate parameter b_{k0} on each of the components τ_k, $k = 1, \ldots, K$ of the observation distribution precisions. As for the Gibbs distribution this will lead to a congugate distribution with independent Gamma posteriors having parameters

$$\tau_k \sim \operatorname{Gamma}(a_k^{VB}, b_k^{VB}), \tag{5.44}$$

where the parameters are given by

$$\begin{aligned} a_k^{VB} &= a_{k0} + N/2, &(5.45a) \\ b_k^{VB} &= b_{k0} + \langle (\mathbf{y}_k - \mathbf{x}_k)^\top (\mathbf{y}_k - \mathbf{x}_k) \rangle_{q(\mathbf{X})}. &(5.45b)\end{aligned}$$

5.3.2 Mean field approximation of the MLFM-SA model

The EM method for the MLFM-SA allowed us to introduce a parameter estimation method consisting entirely of closed form updates during the optimisation procedure. The analytic tractability of the EM method was a direct result of the linear Gaussian dynamic system structure of the successive latent states approximations, this structure allowed us to condition on the states and so write the objective function for the remaining variables as a sum of quadratics. In this section we demonstrate that the same structure carries over to the mean field variational Bayes methods, and so leads to tractable estimation of the approximate distributional estimates.

The conditional independence structure of the MLFM-SA model, as represented in Figure 4.2, naturally suggests using the factorisation

$$q(\mathbf{Z}, \mathbf{g}, \mathbf{B}, \boldsymbol{\tau}, \boldsymbol{\phi}) = q_Z(\mathbf{Z}) q_g(\mathbf{g}) q_B(\mathbf{B}) q_\tau(\boldsymbol{\tau}) q_\phi(\boldsymbol{\phi}),$$

for the approximating distribution. We note that, unlike for the MLFM-AG method, we are able to include a factor for the variable $\boldsymbol{\phi}$. In general there will be no closed form posteriors for the hyperparameters of the latent forces and connection coefficients, and so in this instance we are lead to considering the approximation to the posterior conditional

$$q(\mathbf{Z}, \mathbf{g}, \mathbf{B}, \boldsymbol{\tau}, \boldsymbol{\phi}) \approx p(\mathbf{Z}, \mathbf{g}, \mathbf{B}, \boldsymbol{\tau}, \boldsymbol{\phi} \mid \mathbf{Y}, \boldsymbol{\psi}, \boldsymbol{\zeta}).$$

In what follows we shall suppress the dependence of all density functions on the hyperparameters $\boldsymbol{\psi}$ and $\boldsymbol{\zeta}$ for notational convenience.

The factors of the variational distribution for this model will be given, up to the normalising constants, by the expressions

$$q_Z(\mathbf{Z}) \propto \exp\left\{ \langle \log p(\mathbf{Y}, \mathbf{Z}, \mathbf{g}, \mathbf{B}, \boldsymbol{\tau}, \boldsymbol{\phi}) \rangle_{q_g(\mathbf{g}) q_B(\mathbf{B}) q_\tau(\boldsymbol{\tau}) q_\phi(\boldsymbol{\phi})} \right\}, \tag{5.46a}$$

$$q_g(\mathbf{g}) \propto \exp\left\{ \langle \log p(\mathbf{Y}, \mathbf{Z}, \mathbf{g}, \mathbf{B}, \boldsymbol{\tau}, \boldsymbol{\phi}) \rangle_{q_Z(\mathbf{Z}) q_B(\mathbf{B}) q_\tau(\boldsymbol{\tau})} \right\}, \tag{5.46b}$$

$$q_B(\mathbf{B}) \propto \exp\left\{ \langle \log p(\mathbf{Z}, \mathbf{g}, \mathbf{B}, \boldsymbol{\tau}, \boldsymbol{\phi}) \rangle_{q_Z(\mathbf{Z}) q_g(\mathbf{g}) q_\tau(\boldsymbol{\tau}) q_\phi(\boldsymbol{\phi})} \right\}, \tag{5.46c}$$

$$q_\tau(\boldsymbol{\tau}) \propto \exp\left\{ \langle \log p(\mathbf{Z}, \mathbf{g}, \mathbf{B}, \mathbf{Y}, \boldsymbol{\tau}) \rangle_{q_Z(\mathbf{Z}) q_g(\mathbf{g}) q_\beta(\mathbf{B})} \right\}, \tag{5.46d}$$

$$q_\phi(\boldsymbol{\phi}) \propto \exp\left\{ \langle \log p(\mathbf{Y}, \mathbf{g}, \mathbf{B}, \boldsymbol{\tau}, \boldsymbol{\phi}) \rangle_{q_Z(\mathbf{Z}) q_g(\mathbf{g}) q_\beta(\mathbf{B}) q_\tau(\boldsymbol{\tau})} \right\} \tag{5.46e}$$

For the variational distribution, (5.46d), of the observation precision parameters the form of the optimisation problem is exactly the same as that just discussed for the MLFM-AG method because of the decomposition into an observation model and a structural model. The only difference in this case is that the expectation in will now be taken with respect to the distribution of the variable $\mathbf{Z}^{(M)}$, rather than $\mathbf{X}$, and because of the very close similarities to the result in the previous section we do not provide the full details here.

Mean field update of the latent states' distribution

By using the conditional independence structure of the approximation to the joint density given by MLFM-SA method we have the decomposition

$$\begin{aligned}
\log q_Z(\mathbf{Z}) &= \langle \log p(\mathbf{Y}, \mathbf{Z}, \mathbf{g}, \mathbf{B}, \boldsymbol{\tau}, \boldsymbol{\phi}) \rangle_{q(\mathbf{g}) q(\mathbf{B}) q(\boldsymbol{\tau}) q(\boldsymbol{\phi})} + \text{const.} \\
&= \langle \log p(\mathbf{Y} \mid \mathbf{Z}, \boldsymbol{\tau}) \rangle_{q(\boldsymbol{\tau})} + \langle \log p(\mathbf{Z} \mid \mathbf{g}, \mathbf{B}, \boldsymbol{\Gamma})_{q(\mathbf{g}) q(\mathbf{B})} \rangle + \text{const..} \\
&= \langle \log p(\mathbf{Y} \mid \mathbf{Z}^{(M)}, \boldsymbol{\tau}) + \log p(\mathbf{Y}_\nu \mid \mathbf{Z}_\nu^{(0)}, \boldsymbol{\tau}) \rangle_{q_\tau(\boldsymbol{\tau})} \\
&\quad + \sum_{m=1}^{M} \langle \log p(\mathbf{Z}^{(m)} \mid \mathbf{Z}^{(m-1)}, \mathbf{g}, \mathbf{B}) \rangle_{q_g(\mathbf{g}) q_\beta(\boldsymbol{\beta})} \\
&\quad + \langle p(\mathbf{Z}^{(0)} \mid \boldsymbol{\phi}) \rangle_{q_\phi(\boldsymbol{\phi})} + \text{const..}
\end{aligned} \tag{5.47}$$

As was the case when constructing the conditional distributions for the MLFM-SA in Section 4.3, the Markov property of the set of successive approximations leads to a summation over terms

$$\begin{aligned}\langle \log \mathcal{N}(\mathbf{z}^{(m)} \mid \mathbf{z}^{(m-1)}, \boldsymbol{\Gamma})\rangle_{q_g(\mathbf{g})q_B(\mathbf{B})} = &-\frac{1}{2}\langle (\mathbf{z}^{(m)} - \mathbf{P}\mathbf{z}^{(m-1)})^\top \boldsymbol{\Gamma}^{-1}(\mathbf{z}^{(m)} - \mathbf{P}\mathbf{z}^{(m-1)})\rangle_{q(\mathbf{g})q(\mathbf{B})} \\ &+ \text{const.},\end{aligned}$$

for $m = 1, \ldots, M$. Ignoring the constant term each of these can be arranged as a quadratic form for the augmented vector $\begin{bmatrix}\mathbf{z}^{(m-1)\top} & \mathbf{z}^{(m)\top}\end{bmatrix}^\top$ given by

$$\begin{bmatrix}\mathbf{z}^{(m-1)\top} & \mathbf{z}^{(m)\top}\end{bmatrix} \begin{bmatrix} \langle \mathbf{P}^\top \boldsymbol{\Gamma}^{-1}\mathbf{P}\rangle_{q(\mathbf{g})q(\mathbf{B})} & -\langle \mathbf{P}^\top\rangle_{q(\mathbf{g})q(\mathbf{B})}\boldsymbol{\Gamma}^{-1} \\ -\boldsymbol{\Gamma}^{-1}\langle \mathbf{P}\rangle_{q(\mathbf{g})q(\mathbf{B})} & \langle \boldsymbol{\Gamma}^{-1}\rangle_{q(\mathbf{g})q(\mathbf{B})} \end{bmatrix} \begin{bmatrix}\mathbf{z}^{(m-1)} \\ \mathbf{z}^{(m)}\end{bmatrix}. \tag{5.48}$$

Therefore, the linear dynamical system structure is preserved after taking expectations, and we can determine the parameters of the new transition distribution by identifying (5.48) with the general quadratic form of a linear transition distribution, $\mathcal{N}(\mathbf{z}^{(m)} \mid \mathbf{A}\mathbf{z}^{(m-1)}, \boldsymbol{\Lambda}^{-1})$, which in general takes the form

$$\begin{bmatrix}\mathbf{z}^{(m-1)\top} & \mathbf{z}^{(m)\top}\end{bmatrix} \begin{bmatrix} \mathbf{A}^\top\boldsymbol{\Lambda}\mathbf{A} & -\mathbf{A}^\top\boldsymbol{\Lambda} \\ -\boldsymbol{\Lambda}\mathbf{A} & \boldsymbol{\Lambda} \end{bmatrix} \begin{bmatrix}\mathbf{z}^{(m-1)} \\ \mathbf{z}^{(m)}\end{bmatrix}. \tag{5.49}$$

Identifying the entries of the symmetric matrices in the expressions (5.48) and (5.49) we see that the mean and covariance of the transition densities are solutions to the linear matrix equations

$$\mathbf{A}^\top\boldsymbol{\Lambda}\mathbf{A} = \langle \mathbf{P}^\top\boldsymbol{\Gamma}\mathbf{P}\rangle_{q_g(\mathbf{g})q_B(\mathbf{B})}, \tag{5.50a}$$

$$\mathbf{A}^\top\boldsymbol{\Lambda} = \langle \mathbf{P}\rangle^\top_{q_g(\mathbf{g})q_B(\mathbf{B})}\boldsymbol{\Gamma}^{-1}, \tag{5.50b}$$

where $\langle \mathbf{P}\boldsymbol{\Gamma}^{-1}\mathbf{P}\rangle_{q_g(\mathbf{g})q_B(\mathbf{B})}$ and $\langle \mathbf{P}\rangle_{q(\mathbf{g})q_B(\mathbf{B})}$ are known quantities. Therefore substituting (5.50b) into (5.50a) the matrix $\mathbf{A}$ is given by any solution to

$$\langle \mathbf{P}^\top\rangle_{q_g(\mathbf{g})q_B(\mathbf{B})}\boldsymbol{\Gamma}^{-1}\mathbf{A} = \langle \mathbf{P}^\top\boldsymbol{\Gamma}^{-1}\mathbf{P}\rangle_{q_g(\mathbf{g})q_B(\mathbf{B})},$$

and the, inverse of, the transition covariance matrix is then determined by any solution to

$$\mathbf{A}^\top\boldsymbol{\Lambda} = \langle \mathbf{P}^\top\rangle_{q_g(\mathbf{g})q_B(\mathbf{B})}\boldsymbol{\Gamma}^{-1}.$$

and the, inverse of, the transition covariance matrix is then determined by any solution to

$$\mathbf{A}^\top\boldsymbol{\Lambda} = \langle \mathbf{P}^\top\rangle_{q_g(\mathbf{g})q_B(\mathbf{B})}\boldsymbol{\Gamma}^{-1}.$$

From this we conclude that, after noting the decomposition into a model dependent component and a data dependent component, the same Markov chain distribution structure.

For the data dependent component we need to take the expectations with respect to the approximating distribution for the observation noise and the initial state precision respectively. Using this linear dynamic system structure we have

$$q_Z(\mathbf{z}) \propto \prod_{k=1}^{K} \mathcal{N}(\mathbf{Z}_{\nu k}^{(0)} \mid 0, \phi_k^{-1}) \prod_{m=1}^{M} \mathcal{N}(\mathbf{z}^{(m)} \mid \mathbf{A}\mathbf{z}^{(m-1)}, \mathbf{\Lambda}^{-1})$$

$$\times \prod_{k=1}^{K} \mathcal{N}(\mathbf{Y}_{\nu k} \mid \mathbf{Z}_{\nu k}^{(0)}, \langle \tau_k \rangle_{q(\boldsymbol{\tau})}^{-1}) \mathcal{N}(\mathbf{y} \mid \mathbf{z}^{(M)}, \mathrm{diag}(\langle \boldsymbol{\tau} \rangle_{q(\boldsymbol{\tau})}^{\circ -1}) \otimes \mathbf{I}_N). \tag{5.51}$$

This distribution is clearly Gaussian, but we will rarely have a need to normalise it and so recover the full parameters of the joint distribution. Rather than the parameters of the joint distribution we shall be interested in the evaluation of the expectations

$$\langle \mathbf{\Psi}_0 \rangle_{q(\mathbf{Z})}, \qquad \langle \mathbf{\Psi}_1 \rangle_{q(\mathbf{Z})},$$

where $\mathbf{\Psi}_0$ and $\mathbf{\Psi}_1$ were previously defined by (5.16), and these expectations are most efficiently calculated using Kalman filter methods.

Mean field update of the latent forces' distributions

The update of the latent forces can now proceed in the same manner as the EM algorithm, indeed we shall see that the construction leads to a quadratic similar to that given by (5.18). Using the conditional independence structure the log-density of the variational distribution of the latent forces will be given by

$$\begin{aligned} \log q_g(\mathbf{g}) &= \langle \log p(\mathbf{Y}, \mathbf{Z}, \mathbf{g}, \mathbf{B}, \boldsymbol{\tau}, \boldsymbol{\psi} \rangle_{q_Z(\mathbf{Z}) q_B(\mathbf{B}) q_\phi(\boldsymbol{\phi}) q_\tau(\boldsymbol{\tau})} + \text{const} \\ &= \langle \log p(\mathbf{Z} \mid \mathbf{g}, \mathbf{B}, \boldsymbol{\tau}, \boldsymbol{\phi}) \rangle_q + \log p(\mathbf{g}) \\ &= \langle \log p(\mathbf{z}^{(1)}, \ldots, \mathbf{z}^{(M)} \mid \mathbf{z}^{(0)}, \mathbf{g}, \mathbf{B}) \rangle_{q(\mathbf{Z})} + \text{const}. \end{aligned} \tag{5.52}$$

Using the Markov structure of the successive approximations we can rewrite the first term as the sum of quadratics

$$\begin{aligned} \langle \log p(\mathbf{Z} \mid \mathbf{g}, \mathbf{B}, \boldsymbol{\phi}) \rangle_{q(\mathbf{Z})} &= \sum_{m=1}^{M} \langle \log p(\mathbf{z}^{(m)} \mid \mathbf{z}^{(m-1)}, \mathbf{g}, \mathbf{B}) \rangle_{q(\mathbf{Z})} + \text{const}. \\ &= \sum_{m=1}^{M} \langle \log \mathcal{N}(\mathbf{z}^{(m)} \mid \mathbf{P}\mathbf{z}^{(m-1)}, \mathbf{\Gamma}) \rangle_{q(\mathbf{Z})} + \text{const}. \\ &= -\frac{1}{2} \sum_{m=1}^{M} \langle (\mathbf{z}^{(m)} - \mathbf{P}\mathbf{z}^{(m-1)})^\top \mathbf{\Gamma}^{-1} (\mathbf{z}^{(m)} - \mathbf{P}\mathbf{z}^{(m-1)}) \rangle_{q(\mathbf{Z})} + \text{const}., \end{aligned} \tag{5.53}$$

and so the log-likelihood will have the same basic form as the objective function $\mathcal{Q}_{ode}$ given by (5.17) in Section 5.2.2 where we discussed the EM algorithm, but now not only are we taking the expectation with respect to the variational distribution $q(\mathbf{Z})$, rather than the posterior $p(\mathbf{Z} \mid \mathbf{Y}, \boldsymbol{\theta})$, but we must also take the expectation with respect to the distribution $q(\mathbf{B})$. After including the contribution from the prior the log-likelihood will be a quadratic equivalent to (5.18), and so we may immediately conclude that

$$q_g(\mathbf{g}) = \mathcal{N}(\mathbf{g} \mid \mathbf{m}_g^{VB}, \mathbf{K}_g^{VB}), \tag{5.54}$$

where the parameters of this Gaussian distribution are given by

$$\mathbf{K}_g^{VB} = \left(\langle \mathbf{V}^\top \left(\langle \boldsymbol{\Psi}_0 \rangle_{q(\mathbf{Z})} \otimes \boldsymbol{\Gamma}^{-1} \right) \mathbf{V} \rangle_{q(\mathbf{B})} + \mathbf{C}_\psi^{-1} \right)^{-1} \tag{5.55a}$$

$$\mathbf{m}_g^{VB} = \mathbf{K}_g^{VB} \left[\operatorname{vec}(\boldsymbol{\Gamma}^{-1} \langle \boldsymbol{\Psi}_1^\top \rangle)^\top \langle \mathbf{V} \rangle_{q(\mathbf{B})} - \langle \mathbf{v}_0^\top \left(\langle \boldsymbol{\Psi}_0 \rangle_{q(\mathbf{Z})} \otimes \boldsymbol{\Gamma}^{-1} \right) \mathbf{V} \rangle_{q(\mathbf{B})} \right] \tag{5.55b}$$

Mean field update of the connection coefficients' distribution

The construction of the mean field update for the connection coefficients is given by choosing the alternative representation of the vectorisation of the matrix $\mathbf{P}$, and then using the Markov structure of the state approximations.

As for the latent forces just considered the derivation is largely similar to that already considered for the EM algorithm for the MLFM-SA model, but for completeness we present the full derivation

$$\begin{aligned}
\log q_B(\mathbf{B}) =& \langle \log p(\mathbf{Z} \mid \mathbf{g}, \mathbf{B}) + \log p(\mathbf{B}) \rangle_{q(\mathbf{Z})q(\mathbf{g})} + \text{const.}, \\
=& \sum_{m=1}^{M} \langle \log \mathcal{N}(\mathbf{z}^{(m)} \mid \mathbf{z}^{(m-1)}, \boldsymbol{\Gamma}) \rangle_{q(\mathbf{Z})q(\mathbf{g})} \\
& - \frac{1}{2} \boldsymbol{\beta}^\top \mathbf{C}_\zeta^{-1} \boldsymbol{\beta} + \text{const.} \\
=& -\frac{1}{2} \operatorname{Tr}\left(\langle \boldsymbol{\Psi}_0 \rangle_{q(\mathbf{Z})} \langle \mathbf{P}^\top \boldsymbol{\Gamma}^{-1} \mathbf{P} \rangle_{q(\mathbf{g})} \right) + \operatorname{Tr}\left(\langle \boldsymbol{\Psi}_1 \rangle_{q(\mathbf{Z})} \boldsymbol{\Gamma}^{-1} \langle \mathbf{P} \rangle_{q(\mathbf{g})} \right) \qquad (5.56) \\
& - \frac{1}{2} \boldsymbol{\beta}^\top \mathbf{C}_\zeta^{-1} \boldsymbol{\beta} + \text{const.} \\
=& -\frac{1}{2} \langle \operatorname{vec}(\mathbf{P})^\top \langle \boldsymbol{\Psi}_0 \rangle_{q(\mathbf{Z})} \otimes \boldsymbol{\Gamma}^{-1} \operatorname{vec}(\mathbf{P}) \rangle_{q(\mathbf{g})} \\
& + \operatorname{vec}(\boldsymbol{\Gamma}^{-1} \langle \boldsymbol{\Psi}_1 \rangle_{q(\mathbf{Z})})^\top \operatorname{vec}(\langle \mathbf{P} \rangle_{q(\mathbf{B})}) - \frac{1}{2} \boldsymbol{\beta}^\top \mathbf{C}_\zeta^{-1} \boldsymbol{\beta} + \text{const.} \\
=& -\frac{1}{2} \boldsymbol{\beta}^\top \left(\langle \mathbf{W}^\top \left(\langle \boldsymbol{\Psi}_0 \rangle_{q(\mathbf{Z})} \otimes \boldsymbol{\Gamma}^{-1} \right) \mathbf{W} \rangle_{q(\mathbf{g})} + \mathbf{C}_\zeta^{-1} \right) \boldsymbol{\beta} \\
& + \operatorname{vec}(\boldsymbol{\Gamma}^{-1} \langle \boldsymbol{\Psi}_1 \rangle_{q(\mathbf{Z})})^\top \mathbf{W} \boldsymbol{\beta} + \text{const.}. \qquad (5.57)
\end{aligned}$$

From this we may identify coefficients and once more conclude that the connection coefficients will have a Gaussian distribution given by

$$q(\mathbf{B}) = \mathcal{N}\left(\boldsymbol{\beta} \mid \mathbf{m}_\beta^{VB}, \mathbf{K}_\beta^{VB} \right), \tag{5.58}$$

where the conditional parameters are given explicitly by the expressions analogous to those for the latent forces

$$\mathbf{K}_\beta^{VB} = \left(\langle \mathbf{W}^\top \left(\langle \boldsymbol{\Psi}_0 \rangle_{q(\mathbf{Z})} \otimes \boldsymbol{\Gamma}^{-1} \right) \mathbf{W} \rangle_{q(\mathbf{g})} + \mathbf{C}_\zeta^{-1} \right)^{-1}, \tag{5.59a}$$

$$\mathbf{m}_\beta^{VB} = \mathbf{K}_\beta^{VB} \left[\operatorname{vec}(\boldsymbol{\Gamma}^{-1} \langle \boldsymbol{\Psi}_1^\top \rangle)^\top \langle \mathbf{W} \rangle_{q(\mathbf{B})} - \langle \mathbf{w}_0^\top \left(\langle \boldsymbol{\Psi}_0 \rangle_{q(\mathbf{Z})} \otimes \boldsymbol{\Gamma}^{-1} \right) \mathbf{W} \rangle_{q(\mathbf{g})} \right] \tag{5.59b}$$

Mean field update of the initial state precisions

The variational distribution of the initial state approximation are given by

$$\log q(\boldsymbol{\phi}) = \sum_{k=1}^{K} \langle \log p(\mathbf{Z}_k^{(0)}(\tau_\nu) \mid \phi_k) \rangle_{q(\mathbf{Z})} + \log p(\boldsymbol{\phi}), \tag{5.60}$$

As we discussed when presenting the update for the EM algorithm these parameters were introduced to allow for the approximate inference procedure of Chapter 4, rather than being parameters immediately related to the ODE model. As such they are not well informed by the data using only the observations at a time single point, and so they are better viewed as regularisation parameters because there distribution will be heavily dependent on the choice of prior.

The difference with the parameters $\boldsymbol{\phi}$ in the case of the adaptive gradient matching process is that these parameters now possess a very simple update, by choosing a Gamma prior we will have a conjugate posterior. The simplicity of the log-density (5.60) would allow for much easier sensitivity analysis of the dependence of the inference on these hyperparameters, in contrast the parameters $\boldsymbol{\phi}$ in the MLFM-AG method have no simple posterior unless we place a linear kernel on the interpolating states.

5.4 Discussion

In this chapter, we have used variational methods which better exploit the conditional independence structure of the MLFM to provide efficient methods for obtaining point and distributional estimates. We first offered a demonstration of the EM algorithm for MAP estimation of the parameters in the MLFM using both of the approximations to the posterior distribution of the MLFM we have introduced in this thesis.

For the MLFM-AG method, the relatively simple form of the marginal likelihood first discussed in Section 3.4 means that relatively little benefit is gained from the use of the EM method. We do however note that it would be possible to construct a coordinate based method similar to that considered for the MLFM-SA method, and also very similar to the mean field updates in Section 5.3.1, though in practice there seems to be relatively little benefit to this.

However, the benefit of the EM algorithm for the MLFM-SA method is much more apparent. As discussed in Chapter 4 our approach of completing the model through the use of the set of successive approximations leads to a substantial augmentation of the number of variables, and so optimising the complete likelihood is unfeasible, at the same time the marginal likelihood loses the attractive quadratic structure of the full data model. The EM algorithm allows us to both exploit this quadratic structure and perform a marginalisation step, and because the E-step can be done using Kalman filter methods, it can be done efficiently with a chain that is linear in the order of the approximation.

We then discussed the use of variational Bayesian methods to approximate the intractable posterior of the model. We considered the use of mean fields which factor the posterior as a product of independent components. By choosing a factorisation that naturally respected the conditional independence structures of the model we were able to present Gaussian approximations to the latent force and connection coefficient variables which define the coefficient matrix which we are attempting to learn.

Much of our discussion in Chapter 3 focused on both the interpretation of the parameters state hyperparameters $\boldsymbol{\phi}$ and the regularising parameters $\boldsymbol{\gamma}$ of the gradient expert. Our discussion was mainly focused on both the importance of these parameters on the resulting inference and the fact these parameters have intractable and computationally expensive, conditional posteriors in the MLFM-AG method. This feature is again evident in both the EM and mean field results for the MLFM-AG method where we cannot provide closed-form updates, or mean field factors, for these variables.

In contrast, the hyperparameters of the MLFM-SA model enter the model in a much simpler way, while we have not presented the results in this Chapter both of these parameters can be given a closed form update in the coordinate EM method for the MLFM-SA method and a closed form update for the variational factor. We chose not to present these distributions because of their weak dependence on the data, in general, we believe the role of these regularisation parameters is better assessed as part of a model validation rather than updated as part of the parameter inference process.

As a final remark, we have used the mean field factorisation in Section 5.3.1 and so constructed approximations to the distribution that give (the optimal) independent factors for the latent forces connection coefficients. These variables will not be independent, and so there will be some information loss in this approximation. Our simulation studies in the next chapter suggest that this factorisation still leads to a good estimation of the first moment so that the loss is in the higher-order properties of the model. The mean-field factorisation is not the only way one can construct a variational approximation to the posterior and investigating the possibility of constructing estimates which do not make this full independence assumption may be of interest.

Chapter 6

Simulation studies

6.1 Introduction

In this thesis, we have introduced the MLFM, a flexible model of dynamical systems driven by latent GPs. Unfortunately, parameter inference is complicated by the lack of closed-form expression for the parameter dependent likelihood term. To overcome this, we have introduced the MLFM-AG and MLFM-SA approximations, in Chapter 3 and Chapter 4 respectively, which now provides us with two alternative methods of performing approximate inference for the MLFM. Our introduction of these methods has been accompanied by a discussion of conditions under which we would expect these methods to perform well and in this chapter, we further assess the performance of these methods by evaluating them on synthetic datasets.

In general calculating the exact density is complicated by the lack of an explicit pathwise solution for these models; however, for some simple models, it is possible to approximate the unknown density. This will typically be the case when the Lie algebra supporting the coefficient matrix is trivial, and therefore the matrix exponential in (2.16) is easily inverted. An example of this combination is provided by the Kubo oscillator, a prototypical example of a random harmonic oscillator allowing us to model a dynamical system evolving on the unit circle. This is perhaps the simplest example of the MLFM on a group with nontrivial manifold structure, but still allowing for a simple closed-form expression for the pathwise trajectories. Inverting the pathwise solution allows us to approximate the posterior conditional distribution of the MLFM and we carry out this process in Section 6.2.1. Our analysis in this section allows us to examine the distance between our approximations of the posterior in the space of distributions and in Section 6.2.4 we discuss how this is influenced by geometric properties of the manifold on which the data is constrained to lie. To further aid interpretability we also discuss in Section 6.2.5 how the loss of accuracy in the space of distributions can be translated into more easily interpreted prediction errors in the data space.

The simple example on the circle was made solvable because of the trivial structure of the Lie algebra, in general, we will be interested in higher dimensional models with more complex geometries and so in Section 6.3 we consider fitting the MLFM on rotation valued data. In the absence of a solvable ground truth it becomes hard to evaluate the performance of our methods, and so we consider three performance criteria. The first analysed in Section 6.3.1 is the ability of these methods to reconstruct the observed trajectory, so that the point estimators obtained using our methods display similar properties to the ideal least squares estimators.

The second measure of performance which we examine in Section 6.3.3 is a comparison of the distance of distributional estimates obtained using either the mean field variational approximation to the MLFM-AG model or the MLFM-SA model. Rather than a comparison against a ground truth distribution, which is unknown, this is instead a direct assessment of the consistency of the two different mean field approximations we have introduced in Chapter 5 with one another.

Our final performance measure is an assessment of the difference between the use of Gibbs sampling and the mean field variational approximation which we perform for the MLFM-AG method in Section 6.3.3. Again this is a measure of the consistency of the mean field variational approximation and the MCMC, method and not a measure of closeness to the actual posterior distribution. Nevertheless, knowledge of this consistency combine with the other measures of performance is instructive for when it comes to choosing which particular sampling method to use in a given situation, and we conclude the chapter with a discussion of this and related issues. To futher guide this choice for the practitioner we also provide an extended discussion of the computational complexity of each of these methods in Section 6.3.4. This information concerning the computational complexity, when combined with our results on the loss of accuracy for each of these methods, will allow practitioners to make an informed choice between the methods we have introduced given experimental constraints.

6.2 Simulated dynamic systems on the circle

In this section we consider the class of dynamical systems on the unit circle, these models provide an important test case being both trivially pathwise solvable while at the same time possessing a non-trivial manifold structure. The property of being pathwise solvable will, in turn, allow for a comparison between the ground truth distribution and the approximation methods we are introducing which will be unavailable as we subsequently consider more complex geometries in the remainder of this chapter.

In Section 6.2.1 we shall introduce a prototypical example of an MLFM on the circle, the Kubo oscillator, a model of a random harmonic oscillator which has been of classical interest to physicists. We will present both the pathwise solution of this model and use this to demonstrate an accurate approximation to the conditional distribution of the latent force variables on the basis of a set of observable state variables. Having obtained a suitable approximation to the ground truth distribution we then demonstrate in Section 6.2.3 the accuracy of the approximate inference techniques introduced in Chapter 3 and Chapter 4 using the variational methods introduced in Chapter 5 to recover this true distribution and in particular how the accuracy of this approximation varies under different sampling regimes. In particular, we investigate how the frequency at which samples are collected, and the total time interval we are observing influence the accuracy of the methods which we have introduced.

6.2.1 The Kubo Oscillator

The class of dynamic systems which we will study in this section can be represented by either of two equivalent forms; in the first of

$$\frac{\mathrm{d}z(t)}{\mathrm{d}t} = i[a_0 + g(t)]z(t), \tag{6.1}$$

with $z_0 = z(t_0)$ a point in the complex plane $\mathbb{C}$, typically chosen such that $\|z_0\| = 1$, a_0 is some real valued scalar and $g(t)$ is a smooth real valued Gaussian process. While we introduced the MLFM in Chapter 3 to be real valued the complex form (6.1) makes it immediately clear that a pathwise solution is given formally by

$$z(t) = e^{ia(t-t_0)+i\int_{t_0}^{t} g(\tau)\,\mathrm{d}\,\tau} z(t_0). \tag{6.2}$$

We can recast (6.1) as a real-valued ODE by identifying $\mathbb{C}$ with the plane $\mathbb{R}^2$ which leads us to consider the equivalent system of real-valued ODEs

$$\frac{\mathrm{d}}{\mathrm{d}t}\begin{bmatrix} x(t) \\ y(t) \end{bmatrix} = \begin{bmatrix} 0 & -(a_0 + g(t)) \\ a_0 + g(t) & 0 \end{bmatrix}\mathbf{x}(t). \tag{6.3}$$

Models of the form (6.1) and (6.3) are typical examples of a random harmonic oscillator sometimes referred to as the Kubo oscillator, [Kubo, 2007]. Random harmonic oscillators have been of classic interest [Zoller et al., 1981] where they have been applied to investigate the time-dependent behaviour of the moments of an atomic system under excitation by a laser. As we have discussed in Section 2.3.1 the principal interest in the physics literature has been in deriving qualitative properties of the distribution of the trajectories by marginalising out the latent forces, [van Kampen, 1974], which are typically integrals of a white noise perturbative force. Whereas for the applications we have in mind the force often plays a more interesting role as an exogenous guiding force, and we are then interested in carrying out conditional inference for this unobserved latent force rather than the marginal moments of the trajectory variable.

The expression (6.1) makes it clear that this model is essentially one dimensional, mathematically the Lie algebra is a trivial one-dimensional vector space given by the span of the skew-symmetric matrix in (6.3) and therefore the possibility of noncommutativity does not occur, it, therefore, is straightforward to derive formal solutions for the corresponding MLFM. At this point we shall also absorb the unknown constant a_0 in (6.3) into the Gaussian process term then for the $\mathbb{R}^2$ valued state variable $\mathbf{x}(t) = (x(t), y(t))^{\top}$ we can represent the solution by identifying the complex plane and the real plane and then rewriting (6.2) as the matrix-vector product

$$\mathbf{x}(t) = R\left(\int_0^t g(\tau)\,\mathrm{d}\,\tau\right)\mathbf{x}_0, \tag{6.4}$$

where $R(\theta)$ is the 2×2 real valued matrix which rotates a vector in $\mathbb{R}^2$ by θ-radians anticlockwise around the origin

$$R(\theta) = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix}. \tag{6.5}$$

Because of the closure property of Gaussian processes under linear transformations it follows that the integrated variables $G(ta, t) := \int_{t_a}^{t} g(\tau)\,\mathrm{d}\,\tau$ will themselves be Gaussian random variables with cross covariance

$$\mathbb{E}\left[g(t)G(t_a, t')\right] = \int_{t_a}^{t} k(t,\tau)\,\mathrm{d}\,\tau. \tag{6.6}$$

We now consider the process of carrying out posterior inference for the latent force variable on the basis of a set of observations which we shall assume to be measured with zero noise which have been ordered sequentially with $t_0 < t_1 < \cdots t_N$ which we shall denote by $\mathbf{Y} = (\mathbf{x}(t_0), \mathbf{x}(t_1), \ldots, \mathbf{x}(t_N))$. It then follows from (6.4) that the values of the integrated latent force variables on the successive intervals $G_i := G(t_{i-1}, t_i)$ are constrained by the recursively defined set of solution conditions

$$\mathbf{x}(t_i) = R(G_i)\mathbf{x}(t_{i-1}), \qquad i = 1, \dots, N. \tag{6.7}$$

Because the one parameter family of matrices (6.5) is periodic in the argument θ, the condition (6.7) will suffice to constrain the vector $\mathbf{G} = (G_1, \dots, G_N)^\top$ only up to translation of each of the components by 2π. If we define the vector $\boldsymbol{\gamma} \in [-\pi, \pi]^N$ of principal values defined as

$$\boldsymbol{\gamma} = \{\gamma_i \in [-\pi, \pi] \; : \; \mathbf{x}(t_i) = R(\gamma_i)\mathbf{x}(t_{i-1}), \quad i = 1, \dots, N\}$$

then $p(\mathbf{G} \mid \mathbf{Y})$ has a discrete distribution supported on the infinite lattice

$$\boldsymbol{\gamma} + 2\pi\boldsymbol{\nu}, \qquad \boldsymbol{\nu} \in \mathbb{Z}^N. \tag{6.8}$$

That is to say the posterior of the integrated variables G_i given the data $\mathbf{Y}$ is obtained by constraining the prior of these variables to the disjoint collection of affine spaces defined by the 2π translations. In principle, this would lead to a conditional distribution for the latent force terms which would be an infinite mixture distribution

$$p(\mathbf{g} \mid \mathbf{Y}) \propto \sum_{\nu \in \mathbb{Z}^N} p(\mathbf{g} \mid \mathbf{G} = \boldsymbol{\gamma} + \nu), \tag{6.9}$$

however for reasonably high sampling density and Gaussian process prior with modest variance the Gaussian process $\int_{t_i}^{t} g(\tau)\,\tau$ is going to be almost entirely supported within the interval $[-\pi, \pi]$, this is so because we have the trivial inequality

$$\begin{aligned} \operatorname{Var}\{G_i\} &= \int_{t_{i-1}}^{t_i} \int_{t_{i-1}}^{t_i} k(\tau, \tau')\,\mathrm{d}\,\tau\,\mathrm{d}\,\tau' \\ &\leq \sup_{(s,t)\in[t_{i-1},t_i]^2} |k(s,t)|(t_i - t_{i-1})^2, \qquad i = 1, \dots, N, \end{aligned}$$

so that $\operatorname{Var}\{G_i\}$ is $\mathcal{O}(|t_i - t_{i-1}|^2)$. It follows from this that for reasonably high sampling densities so that the between observation distance $|t_i - t_{i-1}|$ is small for all $i = 1, \dots, N$ we can for all practical purposes consider the approximation obtained by considering only the principal contribution defined by restricting ourselves to conditioning only on the single component of the lattice mixture centered on $\boldsymbol{\gamma}$

$$p(\mathbf{g} \mid \mathbf{Y}) \approx p(\mathbf{g} \mid \mathbf{G} = \boldsymbol{\gamma}). \tag{6.10}$$

Doing so will allow us to consider the conditional distribution as being given by just the single component multivariate Gaussian distribution and since the mean field method of Chapter 5 will lead to multivariate Gaussian distributions, and this allows us to make use of closed-form expressions for the distance between multivariate Gaussian distributions. In the following section, we shall consider the Wasserstein, or Kantorovich-Rubinstein, distance which is a proper metric on the space of probability distributions. For two distributions P and Q the Wasserstein p-distance is defined by

$$d_{W_p}(P, Q) = \inf_{\gamma \in \Gamma(P,Q)} \mathbb{E}_\gamma \left[|\mathbf{x} - \mathbf{y}|^p\right]^{1/p}, \tag{6.11}$$

where for multivariate distributions, P, Q, defined on a common space V this expectation is taken with respect to a "coupling" $\gamma \in \Gamma(P, Q)$ – the space of all distributions on

$V \times V$ such that the marginals are given by P and Q respectively, that is the joint vector $(\mathbf{x}, \mathbf{y})^\top$ has distribution given by γ, while the components $\mathbf{x}$ and $\mathbf{y}$ are distributed according to P and Q respectively, for further details see [Villani, 2008].

A trivial example of such a coupling is the product distribution with independent factors P and Q, however in general the full set of couplings will contain non-trivial joint distributions making the calculation of the expectation in (6.11) challenging. However, for the case of multivariate Gaussian distributions, it can be shown, [Dowson and Landau, 1982], that this is given explicitly by

$$d_{W_2}(P, Q) = \|\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2\|_2^2 + \mathrm{Tr}\left(\boldsymbol{\Sigma}_1 + \boldsymbol{\Sigma}_2 - 2(\boldsymbol{\Sigma}_1^{1/2}\boldsymbol{\Sigma}_2\boldsymbol{\Sigma}_1^{1/2})^{1/2}\right), \tag{6.12}$$

where $\boldsymbol{\mu}_1, \boldsymbol{\Sigma}_1$ represent the mean and covariance matrix determining the distribution P and likewise $\boldsymbol{\mu}_2, \boldsymbol{\Sigma}_2$ denote the mean and covariance matrix of Q, and the norm $\|\cdot\|_2$ is the usual Euclidean 2-norm.

6.2.2 Experimental setup

Having introduced an exactly solvable model we are now in a position to investigate the performance of the two methods we have introduced in this thesis at recovering distributional estimates of the latent force variables using simulated data experiments. As we discussed in Chapter 3 the adaptive gradient matching is based on the use of interpolating processes, and so we might expect it to perform increasingly well as the time span between successive observations decreases because the distribution of the interpolating process becomes better determined and so there is negligible information loss introducing the product of experts assumption.

The successive approximation method of Chapter 4 on the other hand is local. The accuracy of the approximations will likely decrease as we move either forward or backwards in time from the arbitrary initial condition, and as such we might expect the total length of the time interval to be an essential determinant of the method's success, rather than the sampling density. In consideration of these points, we, therefore, chose to carry out our simulation experiments under the two regimes of decreasing sample density and decreasing total sample time range.

For both regimes of interest we simulate the Kubo oscillator (6.3) on an interval $[0, T]$ with N equally spaced observations with temporal distance Δt. We can carry out exact simulation by simulating the integrated variables $\mathbf{G} = (G_2, \ldots, G_N)^\top$, and then form a dataset $\mathbf{Y}$ using the recursive solution given by (6.7). At the same time we may form the vector $\boldsymbol{\gamma}$ in $[-\pi, \pi]^N$ and use this to construct the mean and variance of the approximating Gaussian $p(\mathbf{g} \mid \mathbf{G} = \boldsymbol{\gamma})$ when $\mathbf{g}$ is the N vector of latent forces variables evenly spaced on the interval $[0, T]$. For all experiments we fix the initial value to be $\mathbf{x}_0 = (1, 0)^\top$.

Since our interest is in examining the ability of each method to recover the true latent force we are going to treat the kernel function, and the hyperparameters, as being fixed known quantities. Throughout the following, we will set the kernel function of the single latent force to be the radial basis function kernel with unit parameters given by

$$k_{RBF}(t, t') = \exp\left(-\frac{(t - t')^2}{2}\right).$$

We will also be focusing on the zero noise limit, and so as an approximation, we take the observation noise parameters to be fixed at $\sigma_k^2 = 1e^{-4}$, for $k = 1, 2$. For the remaining hyperparameters and regularising parameters in each method which cannot be directly optimised in the variational framework, we chose to hold these values fixed at their MAP estimates determined by an initial model fit.

The experiments are then carried out by applying both the adaptive gradient method of Chapter 3 and the successive approximation method of Chapter 4 to the simulated dataset using the mean field variational Bayesian method described in Chapter 5. Since by construction the mean field factor, $q(\mathbf{g})$ approximating the conditional distribution $p(\mathbf{g} \mid \mathbf{Y})$ is a Gaussian we will be able to compute exact quantities for the Wasserstein distance using (6.12).

6.2.3 Experimental results

The results for the adaptive gradient matching and successive approximation method are displayed jointly in Table 6.1. Inspecting the column for the MLFM-AG method we see very strong evidence in favour of our hypothesis that it is the gap Δt between observations that have the most significant impact on the performance of this method. The results strongly suggest that the accuracy of the approximation decreases as the space between observations increases. Furthermore, since for fixed Δt the magnitude of the entries in this column is of a similar magnitude for each value of T we conclude that the accuracy of the adaptive gradient matching method is mostly independent of the total size of the interval, and by extension then the number of data points.

The remaining entries of this table report the results for the MLFM-SA model, inspecting each row of these entries we may observe that across all sampling regimes obtained either by varying T, or Δt, that increasing the order of approximation leads to increasingly accurate estimates of the true distribution. Inspecting the columns, we observe that for each order a decrease in the size of the sample window leads to a general increase in the accuracy of the approximation with some local variations within each block of fixed T.

The fact that within most blocks of fixed sample window size that the observed metrics are of a similar magnitude strongly suggests that it is the interval length that is the most critical determinant of the accuracy of the method, rather than the number of sample points or the frequency at which observations are taken. Instead, we observe the phenomena that dense samples can even lead to a slower convergence of the method over longer intervals. This effect is particularly pronounced for the row with $T = 9$ and $\Delta t = 0.50$ which seemingly does an inferior job of approximating the actual distribution at lower orders compared to the sparser samples of the MLFM-SA model, or indeed the MLFM-AG method. It is only after increasing the method to a much higher order that the model begins to perform at a level similar to the other experiments with $T = 9$.

With regards to a direct comparison between the methods we can draw some reasonably general conclusions; at all combinations of the truncation order M we consider the MLFM-AG outperforms the MLFM-SA model when the sampling frequency, $\Delta t = 0.50$, is at its highest. Combined with the analytic and computational efficiency of this method this presents a strong argument for using the MLFM-AG for data observed at a high frequency.

However, as the sampling frequency decreases the additional tuning parameter of the MLFM-SA, and its greater similarity to a numerical solution of the ODE eventually leads to this method outperforming the MLFM-AG method. However, to achieve this superior performance the order of the MLFM-SA method must increase as the total interval length increases.

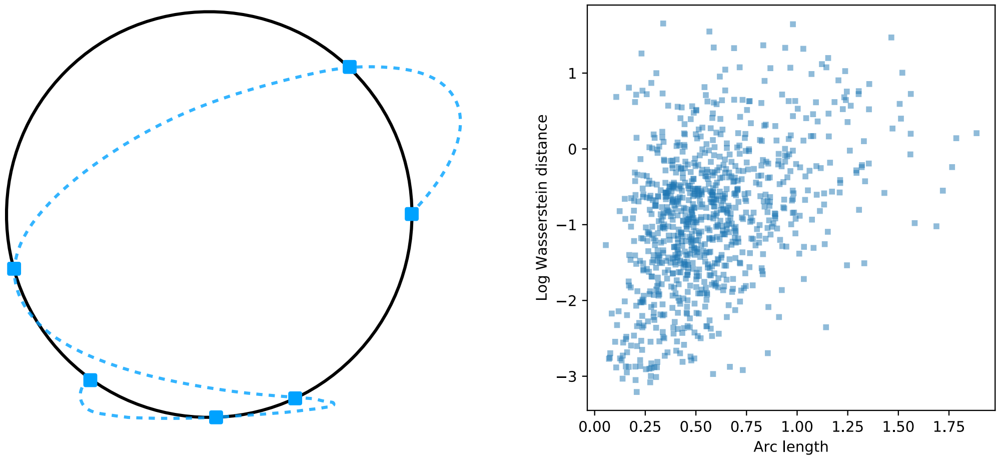
		MLFM-AG	MLFM-SA		
T	Δt		$M = 3$	$M = 5$	$M = 10$
3	0.50	0.179 (0.055)	0.421 (0.190)	0.395 (0.289)	0.191 (0.051)
	0.75	0.328 (0.259)	0.407 (0.411)	0.272 (0.440)	0.076 (0.064)
	1.00	0.727 (0.400)	0.374 (0.311)	0.294 (0.384)	0.185 (0.211)
6	0.50	0.187 (0.075)	0.901 (0.326)	0.557 (0.210)	0.386 (0.209)
	0.75	0.341 (0.247)	0.738 (0.392)	0.629 (0.463)	0.328 (0.325)
	1.00	0.686 (0.662)	0.865 (0.606)	0.619 (0.503)	0.319 (0.412)
9	0.50	0.201 (0.243)	1.517 (0.556)	1.068 (0.450)	0.517 (0.225)
	0.75	0.343 (0.433)	0.983 (0.315)	0.874 (0.407)	0.584 (0.415)
	1.00	0.701 (0.643)	1.051 (0.512)	0.813 (0.522)	0.457 (0.601)

Table 6.1: Results of the simulation study using the mean field variational Bayesian methods to estimate the posterior distribution of the latent forces from the dynamic system (6.3) on the sphere using the MLFM-AG method and the MLFM-SA method of order M. The results display the Wasserstein distance between the true distribution and the mean field approximation. The results display the estimated distance for each setting based on 100 simulations along with an estimate of the standard error.

In summary, the results reported in Table 6.1 suggest that if we have a dense realisation of the trajectory then ideally we would use the adaptive gradient matching method of Chapter 3. Unfortunately this condition is not possible to guarantee, and the MLFM-SA method of Chapter 4 provides the option to overcome a sparsity of sample data at the expense of increased computational costs. While the MLFM-SA performed well with only a low truncation order over small sample intervals, the performance does deteriorate as the interval length increases. It may, therefore, be of interest to replace a single, high order, approximation with a collection of lower order local approximations and combining these local models in a principled manner and we examine the potential of this approach in Section 6.2.6.

6.2.4 Structure loss in the MLFM-AG method

As we have discussed in Chapter 3 gradient matching approaches rely on the interpolating Gaussian process providing a good approximation to the distribution of the true processes, and while as discussed in Section 3.2 the adaptive gradient matching method of [Dondelinger et al., 2013] allows for the system dynamics to better inform this interpolation, it still seems reasonable to expect that the performance deteriorates with the distance between points as ultimately gradient based approaches can only provide local regularisation. Furthermore, it is clear that any breakdown in performance is unlikely to be due to the increasing time between points, but rather the increasing likelihood of a point to drift further away within that time period. Naive interpolation of the process treating them as points in $\mathbb{R}^2$ will not respect the manifold structure and therefore the implied distribution of the trajectory, and by extension its gradient, has the potential to provide a poor approximation when the distance between points is large, where we assume the distance is one that naturally respects the manifold structure, for example the geodesic distance. A graphical representation of this loss in accuracy is displayed in Figure 6.1a. In Figure 6.1b we plot the (logarithm of) the Wasserstein distance of the MLFM-AG approximation from the ground truth distribution as a function of the average arc-length distance between the sample points displaying the deterioration in the approximation as this distance increases.



(a) GP interpolation on S^1

(b) Approximation error vs. arc length

Figure 6.1: (a) Representation of naive Gaussian process interpolation for a process with true support S^1. (b) Monte Carlo estimates of the Wasserstein distance for adaptive gradient matching approximation reported in Table 6.1 as a function of the average arclength between points in the sample

6.2.5 Comparisons of absolute performance

While the Wassterstein distances reported in Table Our results for the Kubo oscillator in Section 6.2.3 enabled us to discuss the relative performances of both the MLFM-AG and the MLFM-SA method as a function of the experimental constraints and order parameters. However, the reported metrics do not provide an intuitive understand of how distance in the approximations to the posterior distribution of the latent force variable will impact the learned representation of our dynamical system, which is ultimately the object we are interested in recovering, and in this section we provide the practitioner with some further diagnostics to better guide this choice.

For a given, finite dimensional, sample of the latent force, $\mathbf{g}$ obtained on uniformly with temporal distance Δt, we denote the solutions obtained by numerically solving the Kubo oscillator ODE (6.3) using a numerical method as $\mathbf{x}(t; \mathbf{g})$ for $t \in \{0, \Delta t, 2\Delta t, \ldots, T\}$. The act of numerically solving the ODE then gives a forward map from samples of the latent force variable to the trajectory space. Since the trajectory space is the one most directly observed by the practioner errors in this space provide a more intuitive diagnostic of how well this method recovers the proper dynamical system. We therefore consider the following measure of forward error

$$\mathrm{RMSE}_{\mathrm{f}}^2 = \frac{1}{DT} \sum_{d=1}^{D} \sum_{n=1}^{T} \mathbb{E}_{P,Q} \left[|x_d(t_n, \mathbf{g}) - x_d(t_n, \mathbf{g}')|^2 \right] \tag{6.13}$$

where $\mathbf{g} \sim P$ and $\mathbf{g}' \sim Q$ and these samples from the respective distributions are assumed independent. In particular we shall be interested in the case where P is the ,

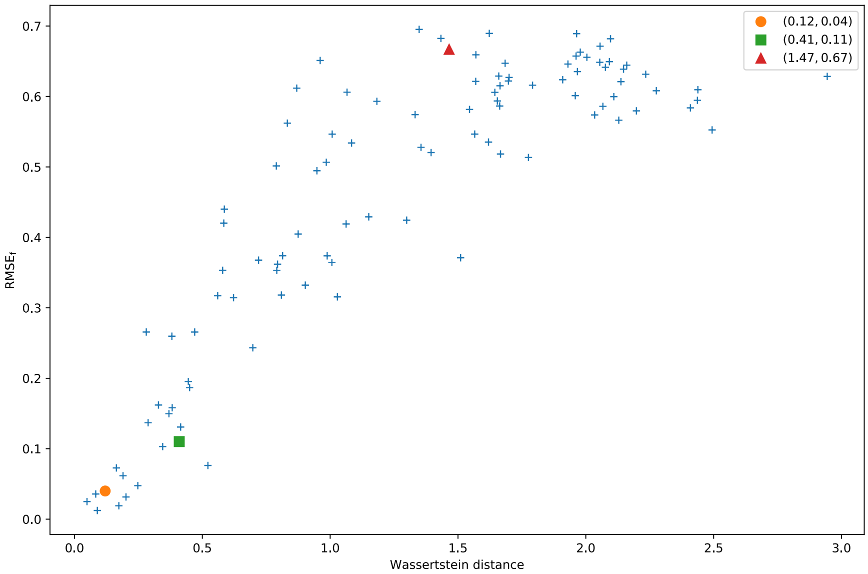
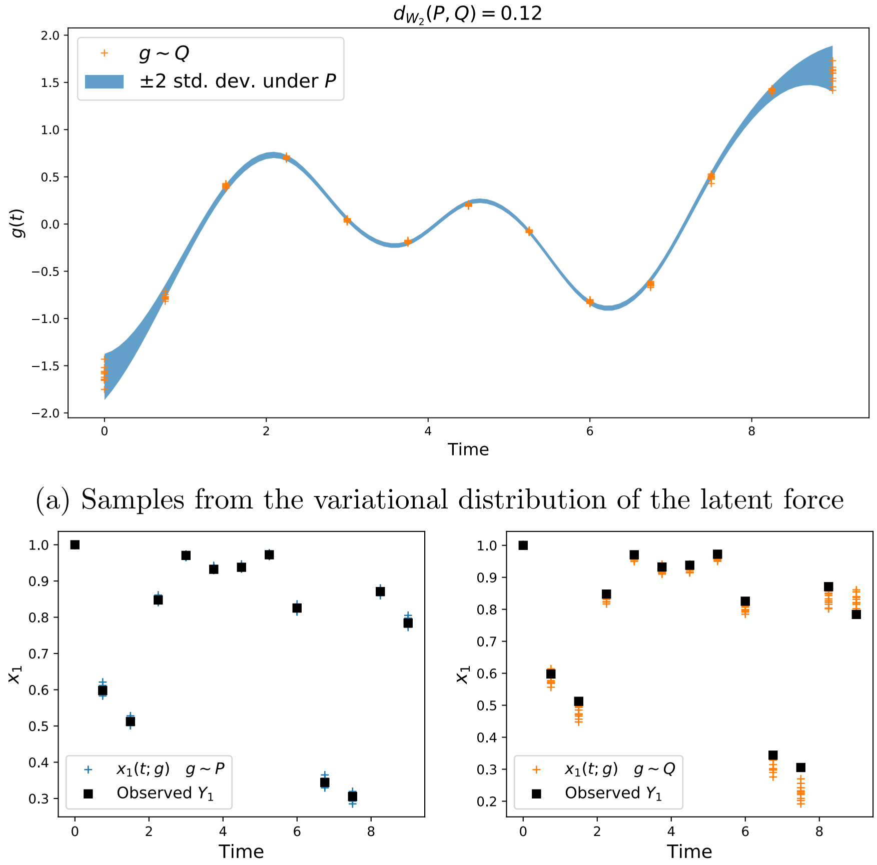


Figure 6.2: Representation of the dependence of the RMSE error (6.13) in data space as a function of the Wasserstein distance between an approximation to the latent force distribution and the true distribution. Results are presented for 100 distinct realisations of the Kubo oscillator on the interval $[0, 9]$ with $\Delta t = 0.75$, and the error (6.13) is estimated using ten samples from the true and approximating distributions respectively.

approximation, to the true distribution described in Section 6.2.1, and Q is one of the variational distributions we have introduce in Chapter 5.

In Figure 6.2 we display results of using Monte-Carlo estimation of (6.13) for a total of 100 different simulations of the Kubo oscillator model on the interval $[0, 9]$ with $\Delta t = 0.75$, the algorithms of the variational approximation were stopped early so that in general the Wasserstein distances are greater than those displayed in Table 6.1 for the same experimental setting, however this wider range in the accuracy of the approximations allows us to gain a broader overview of the relationship between the Wasserstein metric and the RMSE (6.13). The results show that in general as the Wassertstein distance between an approximation of the latent force posterior and the true distribution increases, we see a corresponding increase in the error of the forward map. Furthermore this relationship is approximately linear, with increasing dispersion as the Wasserstein distance increases.

To add further context to the general trends represented in Figure 6.2 we also include visualisations of particular instances of the above experiment, these three particular instances of increasing Wasserstein distance are represented by the '●', '■' and '▲' markers respectively. In Figure 6.3 we display the results for a low value of the Wasserstein distance, $d_{W_2} = 0.12$, in Figure 6.3a we compare samples from the variational distribution with a visualisation of the marginal moments of the true distribution. Even in this case with good agreement of the marginal moments the error in the data space can grow quite quickly, as we see observing Figure 6.3b with 6.3c. Given the accuracy of the mean function most of the reported Wasserstein distance is accounted for by errors in the covariance matrix, since the marginal standard deviation also seems to agreed well with the true distribution this is indicative of the important role of the off-diagonal terms; both in leading to higher values of the Wasserstein distance between multivariate Gaussian distribution, and importantly the *entire* structure of the latent



(a) Samples from the variational distribution of the latent force

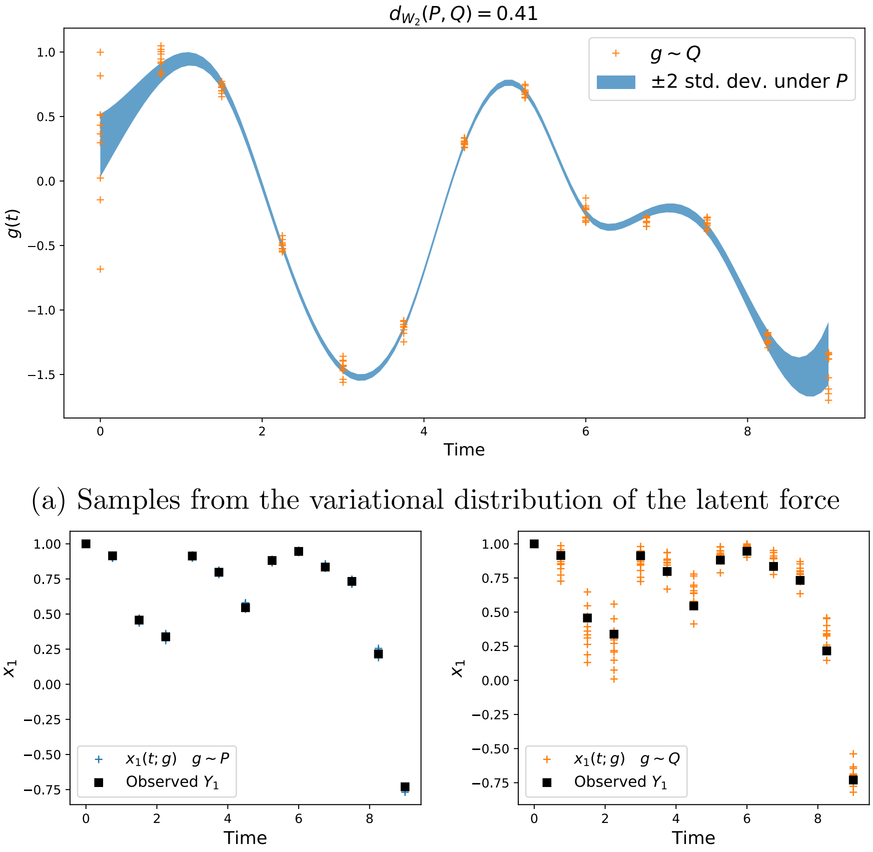
(b) Forward simulation, $g \sim P$

(c) Forward simulation, $g \sim Q$

Figure 6.3: (a) $N = 10$ samples from the variational approximation, Q, of the true posterior distribution, P of the latent force variable for the Kubo oscillator, samples are denoted by '+', also included are the marginal moments of the true distribution P. Comparison of the observations □ forward solution to the ODE using simulated(b) Comparison of forward solutions to the ODE, and the observed data, when the latent force is drawn from the true distribution P. (C) Comparison of forward solutions to the ODE, and the observed data, when the latent force is drawn from the variational approximation Q.

force distribution is when attempting to accurately recover the system dynamics.

As the Wasserstein distance increases to $d_{W_2} = 0.41$ in Figure 6.4 we observe a continuing deterioration in performance, which has become particularly pronounced by Figure 6.5 with $d_{W_2} = 1.47$. For the more modest error in Figure 6.4a we observe that despite the initial over-dispersion the estimated latent force distribution has done a relatively good job at recovering the marginal moments of the distribution, and we see that this initial miss-placed uncertainty leads to higher initial predictive errors in Figure 6.4c, but overall the error is still modest with the forward propagated trajectories still displaying some fidelity to the true dynamics. By way of contrast the results in Figure 6.5a have failed to approximate the true distribution, and the forward propagated samples in Figure 6.5c show little relationship with the true system dynamics.

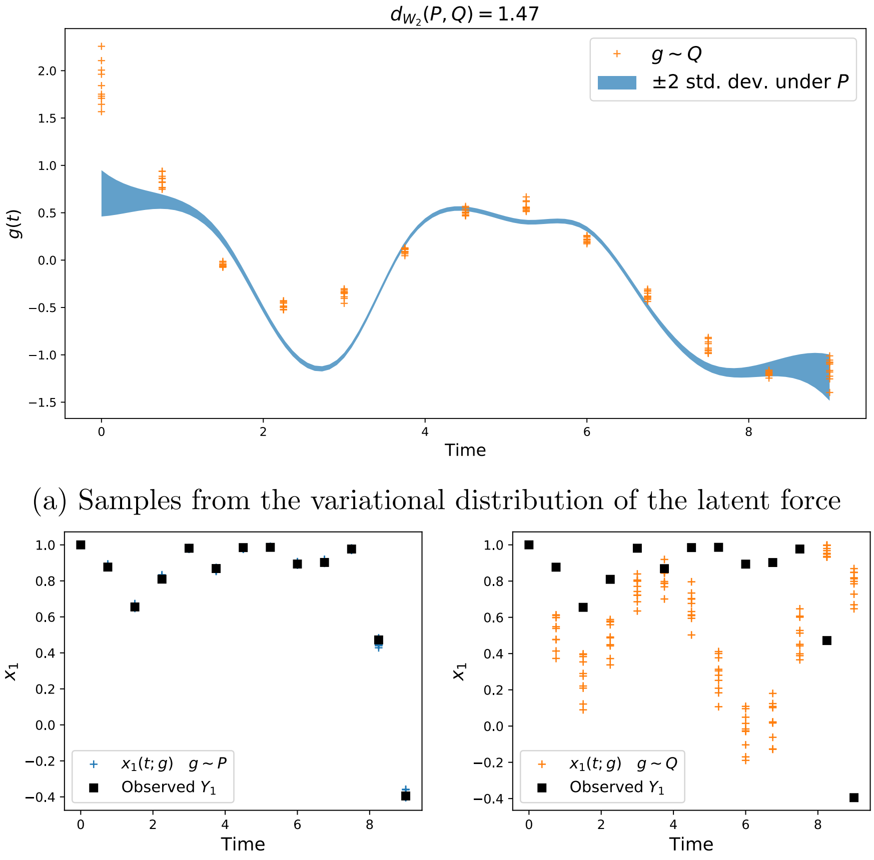


(a) Samples from the variational distribution of the latent force

(b) Forward simulation, $g \sim P$

(c) Forward simulation, $g \sim Q$

Figure 6.4: (a) $N = 10$ samples from the variational approximation, Q, of the true posterior distribution, P of the latent force variable for the Kubo oscillator, samples are denoted by '+', also included are the marginal moments of the true distribution P. Comparison of the observations □ forward solution to the ODE using simulated(b) Comparison of forward solutions to the ODE, and the observed data, when the latent force is drawn from the true distribution P. (C) Comparison of forward solutions to the ODE, and the observed data, when the latent force is drawn from the variational approximation Q.



(a) Samples from the variational distribution of the latent force

(b) Forward simulation, $g \sim P$

(c) Forward simulation, $g \sim Q$

Figure 6.5: (a) Comparison of $N = 10$ samples from the variational approximation, Q, with the marginal moments of the true posterior distribution, P of the latent force variable for the Kubo oscillator experiment, samples are denoted by '+'. (b) Comparison of the observations '■', to the forward solution to the ODE using simulations of the latent force drawn from the true distribution P. (c) Comparison of the observations '■', to the forward solution to the ODE using simulations of the latent force drawn from the variational approximation, Q.

6.2.6 MAP estimates of the MLFM-MixSA model

In Section 4.5 we considered the possibility of combining several local models of a lower order, rather than a single model of a possibly higher order. The hope being that by introducing local experts to the solution we can alleviate the deteriorating performance observed in Table 6.1 for the MLFM-SA model as the total length of the sampling interval increases. In doing so, we will now have two adjustable integer tuning parameters; the number of mixture centres, Q, and the order of the approximation, M.

In this section, we shall repeat the experimental setup used described in Section 6.2.2. Rather than investigating the distance in distribution, we compare the ability of the MLFM-MixSA model to recover the MAP point estimates of the latent forces as implied by the Gaussian approximation (6.10). We consider up to three mixture components. For the case $Q = 1$ we place a single element centred at the midpoint of the interval, for $Q = 2$ we set one component at the initial value and one at the end of the range. Finally, for $Q = 3$ we place a mixture centred at the beginning and end of the interval, along with one centred on the midpoint of the interval. Finally we use softmax functions of the form (4.51) to model the mixture probabilities.

To investigate the model, we shall fix the order of the approximation at $M = 5$, and examine the accuracy of the MLFM-MixSA method as we vary both the experiment settings and the number of mixture components. Referring back to the results in Table 6.1 the results for $M = 5$ in the single component case where accurate for short interval lengths, but quickly became inaccurate for the longer intervals.

The results are displayed in Table 6.2. If we consider first when the number of mixture components is set at $Q = 1$, i.e. we are considering the regular MLFM-SA model, then the results show a similar pattern to those in Table 6.1 with decreasing performance as the total interval length increases. While the pattern is similar, in general, the results for the MAP estimates in this study will be lower because we are only examining the accuracy of the point estimates rather than the properties of the full distribution.

If we examine the shortest interval length, $T = 3$, then increasing the number of mixture components shows no apparent benefit in the performance in the model. Our results show a slight increase in the estimated distance from the real value, although this is accompanied with a higher estimated error, possibly due to the more complicated optimisation of the mixture model with the increases parameterisation of the higher component model.

As the entire sampling interval increases the benefit of the mixture modelling approach becomes more evident, and this is particularly noticeable for the most prolonged time interval considered, with the change from the single model to the two-component involving a significantly improved performance, and this being true for all the considered sampling frequencies.

The results reported in this section are suggestive of the potential for improving the performance of the MLFM-SA method if we account for the inherently local structure of the approximation. However, we still view the problem of doing this in the most appropriate way as an open problem and return to discussing it in Section 8.2.2.

6.3 Simulated dynamic systems on $SO(3)$

An important feature of the MLFM framework we have introduced is the ability to both model and learn, the action of a group on a vector space. Important to this process will be the ability to learn the coefficient matrix of the IVP corresponding to

		No. of mixtures		
T	Δt	$Q = 1$	$Q = 2$	$Q = 3$
3	0.50	0.184 (0.177)	0.207 (0.216)	0.255 (0.231)
	0.75	0.163 (0.175)	0.231 (0.243)	0.252 (0.280)
	1.00	0.207 (0.214)	0.193 (0.265)	0.291 (0.315)
6	0.50	0.232 (0.281)	0.187 (0.217)	0.211 (0.322)
	0.75	0.276 (0.255)	0.158 (0.178)	0.133 (0.228)
	1.00	0.410 (0.381)	0.104 (0.210)	0.153 (0.244)
9	0.50	1.109 (0.400)	0.247 (0.272)	0.209 (0.366)
	0.75	0.751 (0.501)	0.388 (0.261)	0.133 (0.241)
	1.00	0.981 (1.133)	0.440 (0.393)	0.164 (0.322)

Table 6.2: Results of the simulation study using the MLFM-MixSA model with Q mixtures to recover the MAP estimate of the latent forces from the dynamic system (6.3) on the sphere. The results display the mean and standard error of the euclidean distance between the true MAP value of the latent forces, and those obtained optimising the MLFM-MixSA model using 100 simulations of $N = T/\Delta t + 1$ observations from the model on $[0, T]$.

the fundamental solution

$$\frac{\mathrm{d}\mathbf{x}(t)}{\mathrm{d}t} = \mathbf{A}(t)\mathbf{x}(t), \qquad \mathbf{x}(t_0) = \mathbf{I}_K. \tag{6.14}$$

This differs slightly from the introduction of the MLFM earlier in this thesis as the state variable will now be matrix valued, we can optionally vectorise the state variable leading to the vectorised variant of the time-evolution law

$$\operatorname{vec}(\mathbf{A}(t)\mathbf{x}(t)) = (\mathbf{I}_K \otimes \mathbf{A}(t)) \operatorname{vec}(\mathbf{x}(t)),$$

which is now in the usual form of our MLFM with coefficient matrix $\mathbf{I}_K \otimes \mathbf{A}(t)$. This construction, however, leads to redundant zero entries in the new coefficient matrix, and an unnecessary increase in the dimension of the state variable. More efficient is to model the system as K independent observations from the IVP

$$\dot{\mathbf{x}} = \mathbf{A}(t)\mathbf{x}(t), \qquad \mathbf{x}_0 = \mathbf{e}_k,$$

where $\mathbf{e}_k$ is the canonical basis vector of $\mathbb{R}^K$, for $k = 1, \ldots, K$. Specifying the model to include multiple independent outputs in this way requires only minor adaptations of the expressions presented in the previous chapters.

In this section we shall consider the case where the group in question is given by the collection of proper, i.e. orientation preserving, rotations of vectors in $\mathbb{R}^3$, this group is typically denoted by $SO(3)$. To construct a random trajectory on this group we constrain $\mathbf{A}(t)$ in (6.14) to be supported on the Lie algebra $\mathbf{so}(3)$ of 3×3 skew-symmetric matrices with trace zero. This is a three dimensional vector space with the canonical choice of basis $\{\mathbf{L}_d\}_{d=1}^{3}$ given by

$$\mathbf{L}_1 = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & -1 \\ 0 & 1 & 0 \end{bmatrix}, \qquad \mathbf{L}_2 = \begin{bmatrix} 0 & 0 & 1 \\ 0 & 0 & 0 \\ -1 & 0 & 0 \end{bmatrix}, \qquad \mathbf{L}_3 = \begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}.$$

The example on the circle considered in the previous section may be identified with the group $SO(2)$ and so the model in this section may be considered as the higher dimensional analogue to the Kubo oscillator model analysed above. However, since the Lie algebra is non-trivial, it will not be possible to present a pathwise solution to this model analogous to (6.7). Therefore a comparison between our introduced approximations and some ground truth summary statistic is no longer possible, and this greatly complicates the validation of the approximate methods we have introduced.

In the absence of known ground truth summaries of either point estimates or distributional summaries, we propose the following indirect measures of performance for the methods we have introduced.

The first of these reflects the ideal scenario we would observe if we were able to use a nonlinear least squares and numerical integration of the ODE to obtain the model parameters. By minimising the squared error the estimates obtained using numerical integration will closely interpolate the observed data, and as such it is desirable that when the estimates obtained using the methods we have introduced are used as inputs to a numerical integration that the resulting trajectory has a small squared error. We refer to this measure as the *reconstruction error* and discuss it in more detail in Section 6.3.1.

While the first measure is intended to assess the accuracy of our approximations in recovering the ground truth dynamics, the criteria (B) and (C) are not intended to represent comparisons with any ground truth values, and instead are measures of internal consistency of our approximations to the MLFM.

Criteria (B) is a measure of whether the approximations return similar estimates of the posterior distributions of the model variables. Since both the MLFM-AG and the MLFM-SA method return variational approximations with Gaussian factors so we can report the Wasserstein distance between the two approximations. When these distances are small, we can conclude that these approaches produce similar estimates, and so favour the more efficient method. On the other hand when this distance is larger the use of one method will give more accurate results than the other, and we can carry out this selection with reference to the reconstruction error and our results for the Kubo oscillator.

Our final criteria assess whether the estimates obtained using Gibbs sampling are similar to those obtained using the mean field variational methods. In particular, the mean field variational approximation involves simplifying independence assumptions for the posterior which are not necessary for the MCMC methods. However, the deterministic approximations are much more efficient, and so the analysis in this section will enable us to understand better what we must sacrifice in accuracy for this efficiency. We only carry out this assessment for the MLFM-AG model because, as discussed in more detail in Section 4.6 the structure of the MLFM-SA model is less well suited to Gibbs sampling.

(A) Accuracy of the trajectory implied by obtained point estimates.

(B) Between method consistency of distributional estimates obtained using either the MLFM-AG or the MLFM-SA methods.

(C) Consistency of summary statistics obtained using MCMC methods and those obtained using variational methods.

Experimental setup

For all of the experiments reported in this section we treat the observations as noise-free, and so in what follows we approximate this by taking the parameters σ_k^2 to be fixed at $1e^{-4}$ for $k = 1, \ldots, K$. As for the simulation study of the Kubo oscillator, we will only consider a single latent force, $R = 1$, but now allow the coefficients β_{rd} to vary freely. For the connection coefficients, we assign the i.i.d. priors of the form

$$\beta_{rd} \sim \mathcal{N}(0, 1),$$

for $r = 0, 1$ and $d = 1, 2, 3$. For the single latent force, we again use an RBF kernel with unit hyperparameter, and this is assumed known during the modelling process. We further note that as long as $[\mathbf{A}_0, \mathbf{A}_1] \neq \mathbf{0}$ then these pair of matrices generate the complete Lie algebra $\mathfrak{so}(3)$, so that even with a single latent force this model is still controllable, see Example 8.1 in [Jurdjevic and Sussmann, 1972].

For each experimental setting defined by the pair $(T, \Delta t)$ we simulate 100 observations from the prior and report the mean and standard deviation of our given summary measure. Unlike in the previous section we cannot construct the pathwise solution for this model, and so it will not be possible to perform exact simulation of the observations. We, therefore, propose to simulate from the model using a densely simulated path of the latent GP, solving the ODE using numerical methods evaluated at these dense points, and then downsampling the dense time series. By a dense sample path realisation of a GP, we mean a suitably fine realisation so that there is negligible conditional variance between the partition nodes. As for our previous experiment, we will assign each GP with an RBF kernel with unit length scale parameter, and we chose a uniform partition of the interval with spacing $1e^{-3}$.

6.3.1 Reconstruction error

In models with a non-trivial Lie algebra when we can no longer provide, even an approximation to, the ground truth MAP estimates then model assessment becomes much more challenging. Even though we have access to the actual functions used to simulate these processes, there is no good reason to believe that even the true, but unavailable, MAP estimates should be close to the generating function in a pointwise sense for sparsely sampled data.

With this in mind, we consider an alternative point estimate which we refer to as the *reconstruction error at the MAP estimate.* The motivation being that more critical than any pointwise estimate of the function is the ability of our procedure to return an estimate of the coefficient matrix that does a good job of interpolating through the observed data.

This measure is also appealing because of its similarity to properties which would be observed for estimates obtained by using least squares methods as discussed in Section 1.1.1. While owing to the infinite dimensional GP parameters, we have discounted the possibility of obtaining parameter estimates by using numerical integration if it were possible for parameter estimates to be obtained using this method, and a global minimum achieved then for a well-specified model the implied trajectories obtained by numerically solving the ODE should closely interpolate the observed data. Our reconstruction error is designed to measure how closely the implied trajectory at the MAP estimate using our approximation fits the observed data.

We construct this estimate by first fitting the model using the methods discussed in Chapter 3 or Chapter 4 to jointly obtain a pair of MAP estimates $(\hat{\mathbf{g}}, \hat{\mathbf{B}})$ from which we then form an estimate of the component functions of the matrix $\mathbf{A}(t)$. As discussed above we treat the parameters of the observation noise and the GP hyperparameters as fixed, and we do not report the estimates of these or any remaining model specific parameters. We then numerically solve the MLFM system using the estimated function and report the pointwise distance of the solution and the estimated values

$$\hat{RE} = \frac{1}{NK} \sum_{i=1}^{N} \sum_{k=1}^{K} (\mathbf{Y}_{ik} - \hat{x}_k(t_i))^2,$$

where $\mathbf{x}(\hat{t_i}) = \mathbf{x}(t_i; \hat{\theta})$, is the trajectory obtained by numerically solving the model with the MAP estimates of the parameters and the known initial conditions. The reconstruction error is, therefore, the mean squared error obtained solving the model using the MAP estimates. By reporting pointwise distances of the implied solution, rather than pointwise distances in the function space, we construct a measure which better reflects the fidelity with which either method has learned a good estimate of the coefficient matrix, and therefore the driving dynamics.

The synthetic data is generated using the general approach outlined above. We construct the MAP estimates for the MLFM-AG method using the direct optimisation of the marginalised density as in Section 3.4, and for the MLFM-SA method we use the EM algorithm described in Section 5.2.2.

The results are displayed in Table 6.3. The conclusions are broadly similar to those obtained for the similar experiment we performed using the Kubo oscillator. The MLFM-AG method appears to demonstrate accurate parameter estimates which interpolate the observed data well for high sampling frequencies, but this performance begins to deteriorate as the distance between observations increases. Once more this phenomena is likely related to information loss in the interpolating process.

For the MLFM-SA method, we again observe the requirement of increasingly higher orders of the expansion to achieve accurate results as the total length of the sampling interval increases. However, with sufficiently high order this method does perform equivalently to the MLFM-AG method for small values of Δt and has the benefit of being adjustable so that it can be used in regions where the MLFM-AG method performs poorly.

For both methods there is a more pronounced dependence in total sampling interval length, this is expected to the MLFM-SA method, but the results for the MLFM-AG method in Table 6.1 were mostly independent of dimension, and so this is more surprising for this method. Whether this is a genuine feature of the model in higher dimensions or an artefact of the chosen measure of fit is hard to say definitively in the absence of a ground truth value. However, it is highly plausible that this is an artefact of the reconstruction error, which by solving the ODE from a given initial condition is likely to display more obvious drift away from the accurate trajectories over longer time intervals, rather than a systematic feature of the MLFM-AG approximation.

		MLFM-AG	MLFM-SA		
T	Δt		$M = 3$	$M = 5$	$M = 10$
3	0.50	0.015 (0.105)	0.341 (0.132)	0.098 (0.101)	0.021 (0.113)
	0.75	0.012 (0.161)	0.442 (0.281)	0.023 (0.181)	0.017 (0.109)
	1.00	0.221 (0.191)	0.790 (0.619)	0.231 (0.230)	0.098 (0.113)
6	0.50	0.031 (0.131)	0.983 (0.410)	0.499 (0.324)	0.024 (0.107)
	0.75	0.125 (0.174)	1.319 (0.723)	0.393 (0.289)	0.103 (0.156)
	1.00	0.319 (0.253)	1.101 (0.612)	0.632 (0.312)	0.143 (0.132)
9	0.50	0.061 (0.138)	0.881 (0.415)	0.321 (0.261)	0.093 (0.146)
	0.75	0.214 (0.213)	1.005 (0.489)	0.466 (0.235)	0.191 (0.273)
	1.00	0.361 (0.268)	1.181 (0.521)	0.761 (0.419)	0.399 (0.284)

Table 6.3: Reconstruction error after solving the ODE (6.14) on the interval $[0, T]$ using the MAP estimates conditioned on $N = T/\Delta t + 1$ equally spaced observations. The MAP estimates were obtained via the MLFM-AG method and the MLFM-SA method with truncation order M.

6.3.2 Comparison of the mean field estimates

In this thesis, we have introduced two methods for approximating the distribution of the model parameters, and in Chapter 5 we, described the mean field variational approximation for the MLFM-AG and the MLFM-SA method respectively.

Our discussion of both of these methods has indicated there are specific scenarios under which we would expect one method to perform well relative to the other. In this section, we compare the variational distributions estimated by both of these methods under the different experimental settings. Once again we must emphasise that in the absence of a ground truth approximation to the distribution this test only represents a test of whether these two approaches lead to similar, potentially incorrect, inference. This will still provide useful information for choosing an approximation in a given situation when combined with our results for the Kubo oscillator and the reconstruction error in the previous section.

We carried out the experiment using the approach described in Section 6.3, and the results are displayed in Table 6.4. If we first consider the regions of low Δt, but high values for the order parameter M, then we observe that the distance between the two approximations is comparatively small. This suggests that both methods are consistent with one another, and given our previous results it is reasonable to hope that they are both also close to the exact unknown distribution. On the other hand, if we consider the case where Δt is large, and M is small or vice versa we observe the distance between the estimated distributions are large. This, combined with our previous results, would support the claim that under these experimental settings one method is displaying reasonably accurate results while the other is performing poorly.

The interpretation of our results is less clear for the remaining experimental settings; under these regimes, our results from the Kubo oscillator experiment would suggest both methods are likely to perform only moderately well. However, what the results in this experiment further indicate is that these methods do not deteriorate consistently with respect to one another. That is they estimate distributions which are unlikely to be close to the actual distribution but also are not close to one another. Under these scenarios, the MLFM-SA method has the advantage of being able to increase the order of the approximation and so perhaps better recover the actual distribution, but at a higher computational cost.

In summary, the results suggest that when we have access to data observed at a high sampling frequency, we will get similar results if we use the MLFM-AG or the MLFM-SA method with a high enough approximation order relative to the length of the sample interval. In those circumstances when we have less control of the frequency with which data is observed the option of increasing the order parameter in the MLFM-SA method may be preferable, however, this comes with an increase in the computational complexity. Of course, these results only indicated the consistency of the variational approximation, and not whether this approximation is similar to the true distribution or that which would be realised by sampling from the posterior using MCMC methods and so we consider this aspect of the problem in the next section.

T	Δt	$M = 3$	$M = 5$	$M = 10$
3	0.50	1.368 (0.415)	0.955 (0.347)	0.231 (0.151)
	0.75	1.076 (0.394)	1.110 (0.315)	0.974 (0.331)
	1.00	1.051 (0.359)	1.221 (0.370)	1.117 (0.415)
6	0.50	1.259 (0.320)	0.874 (0.387)	0.187 (0.113)
	0.75	1.423 (0.585)	0.691 (0.511)	0.477 (0.411)
	1.00	1.113 (0.383)	1.042 (0.402)	1.131 (0.608)
9	0.50	1.430 (0.465)	1.334 (0.401)	0.318 (0.287)
	0.75	1.317 (0.501)	1.256 (0.487)	0.828 (0.511)
	1.00	1.719 (0.612)	1.827 (0.431)	1.242 (0.429)

Table 6.4: Wasserstein distance between the mean field distribution of the variable $(\mathbf{g}, \boldsymbol{\beta})^\top$ calculated using either the mean field factorisation of the MLFM-AG method, or the mean field factorisation of the MLFM-SA model with order M. Reported are the mean and standard deviation of the distance obtained from 100 simulations of $N = T/\Delta t + 1$ equally spaced observations of the process on $[0, T]$.

6.3.3 Consistency of the MLFM-AG method

We now turn to an examination of point (C) in Section 6.3. This study will allow us to assess under what conditions the estimated distribution using the MLFM-AG approximation is mostly independent of the method used to estimate it. Given the advantages that deterministic methods such as the mean field variational Bayes methods often have in terms of computational efficiency compared to sampling methods, it would be useful to know under what conditions these methods give equivalent results to that obtained using MCMC methods, which have the advantage of more readily available guarantees of simulating from the proper posterior.

As discussed in the introduction to this section we will only consider the performance for the MLFM-AG method. For each experiment we obtain 100 observations of the pair $(\mathbf{g}, \mathbf{B})$ from the posterior using Gibbs sampling from the conditional distributions presented in Chapter 3. Before collecting samples, we run the sampler with a burn-in period of 500 iterations of the chain. These values are chosen to balance the length of time taken to run these experiments while allowing the chain to converge to the posterior. The variational estimates are obtained by updating the optimal mean field factors as described in Chapter 5. For both methods, the hyperparameters and regularisation parameters are held fixed at initial values determined by running the MAP estimation routine.

While the conditional distribution of the latent force and connection coefficient variables are Gaussian, there is no reason to believe the marginal posteriors will be, and therefore we cannot report a proper estimate of the distance between a distributional estimate from the MCMC observations, and the variational factors. Instead, we chose to take as an approximation the Gaussian distribution with moments matching those observed in the MCMC sample and then report the Wasserstein distance, while this is necessarily an approximation it is still an illuminating one. To further clarify the difference between the two approximations we decompose the Wasserstein 2-distance (6.12) between two normal distributions, P_1 and P_2, as

$$d_{W_2}(P_1, P_2) = d_{\mu}(P_1, P_2) + d_{\Sigma}(P_1, P_2),$$

where

$$\begin{aligned} d_{\mu}(P_1, P_2) &= \|\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2\|_2^2, \\ d_{\Sigma}(P_1, P_2) &= \operatorname{Tr}\left(\boldsymbol{\Sigma}_1 + \boldsymbol{\Sigma}_2 - 2(\boldsymbol{\Sigma}_1^{1/2}\boldsymbol{\Sigma}_2\boldsymbol{\Sigma}_1^{1/2})^{1/2}\right). \end{aligned}$$

This further decomposition into a mean dependent term and a covariance matrix dependent term will allow us to assess to what degree the difference in the estimated distribution arises from errors in the location of the point estimates, or if errors in the second moment primarily drive it.

One immediate implication of our choice to approximate the distribution of the MCMC sample using its sample moments is the fully connected covariance matrix for the forces and connection coefficients, in comparison, the distribution of these variables under the mean field approximation is necessarily independent. Therefore the results presented in this section will allow us to quantify the information loss arising from the independence assumption.

The results are displayed in Table 6.5. If we again first inspect the regions in which we expect both methods to perform well then we observe they are also producing estimates of the distribution which are close to one another. This is especially true when inspecting the difference between the estimated means, and it seems reasonable to suppose these methods are close to recovering the true mean function.

For those regions in which our previous results suggest that in general the MLFM-AG method is likely to give inaccurate results, we also see a disagreement in the variational and MCMC distributions, and this is particularly true of the second order moments. When the distance between the observed data points is larger, there is a corresponding increase in the volume of possible forces and connection coefficients which would interpolate a given set of points, and these parameters are naturally correlated with one another. With this in mind, the additional information loss in the second order structure above that already experienced in the location estimate is not surprising. However, since in general, we would be reluctant to suggest this method under these settings it is questionable how important this lack of consistency is.

In general, and even for those methods in which we would expect the method to be performing well, our distance measure for the second-order moments, d_{Σ}, seems to offer some evidence to suggest that these methods lead to different approximations of the posterior distribution and a more pronounced accuracy loss for the higher order moments. However, this distance is slight for frequently observed data, and given the significant advantages in computational efficiency, there are good reasons to prefer the mean field methods. A deeper understanding of the relative performance of these two methods is therefore an appropriate direction for future research.

T	Δt	d_μ	d_Σ
3	0.50	0.091 (0.069)	0.114 (0.126)
	0.75	0.208 (0.193)	0.576 (0.278)
	1.00	0.272 (0.433)	0.839 (0.877)
6	0.50	0.103 (0.077)	0.137 (0.185)
	0.75	0.189 (0.211)	0.613 (0.355)
	1.00	0.281 (0.244)	0.881 (0.701)
9	0.50	0.087 (0.081)	0.126 (0.199)
	0.75	0.221 (0.255)	0.628 (0.401)
	1.00	0.331 (0.411)	0.771 (0.774)

Table 6.5: Results of the simulation study of the mean and covariance of the distributions obtained using the MCMC approximation and the variational approximation of the MLFM-AG model. Reported are the mean and standard error of the distance between the estimated mean and covariance functions for the pair $(\mathbf{g}, \boldsymbol{\beta})^\top$ obtained using 100 simulations of $N = T/\Delta t + 1$ observations from the model on $[0, T]$.

6.3.4 Computational complexity

The principle motivation for introducing the variational approximations for both the MLFM-AG and MLFM-SA method in Chapter 5 was the claim that deterministic methods are able to achieve much greater computational efficiency than the corresponding sampling based methods, and we now provide some supporting evidence for this claim.

If we first consider the computational complexity of the variational methods, then from Section 5.3.1 we see that the principal computational complexity per-iterate for the MLFM-AG method is in computing the inversions $\mathbf{C}_{\boldsymbol{\phi}_k}^{-1}$ and $\mathbf{S}_{\boldsymbol{\phi}_k}^{-1}$ which correspond to the prior GP model on the states and the marginalisation over the gradient process respectively. The dimension of each of these matrices is given by the number of data points, N, and therefore we can conclude that the computational complexity per-iterate for the MLFM-AG method behaves like $\mathcal{O}(N^3)$ with N the number of data points.

However, for the MLFM-SA method there are two additional points which must be considered; first is the path augmentation setup described in Section 4.3, which means we must consider the inversion of $\tilde{N}$, matrices with $\tilde{N} > N$. For the computational complexity we then also have to consider the computational complexity of the Kalman filter approximation to the successive approximations when calculating the mean field updates for the latent states' distribution as described in Section 5.3.1. This is linear in the chain length, but each step is cubic in the state dimension [Säarkä, 2013], as such that computational complexity of the MLFM-SA method which has been augmented to $\tilde{N}$ states, and with the approximation taken to the order of M has a per-iterate complexity of $\mathcal{O}(\tilde{N}^3 M)$, with $M \ll \tilde{N}$. We can conclude that the MLFM-SA method with just a single order of approximation will have a higher per-iteration time than thee MLFM-AG method for the same number of datapoints, and that the time taken to use this method will further scale linearly when increasing the approximation order. We demonstrate this theoretical complexity empirically by simulating 100 trajectories of the model constrained to SO(3) on the interval $[0, T]$ with $T \in \{6, 9\}$, and then performing

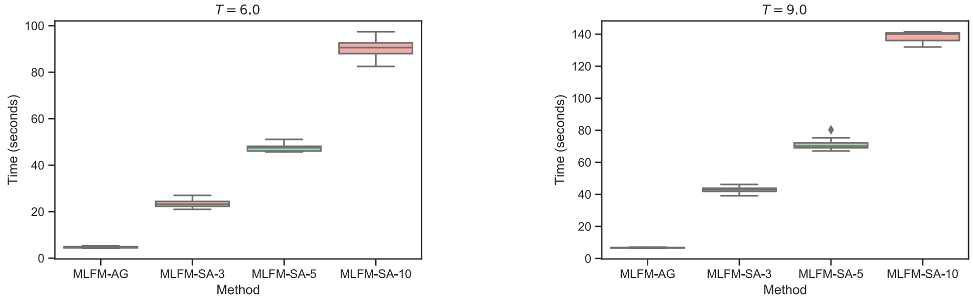
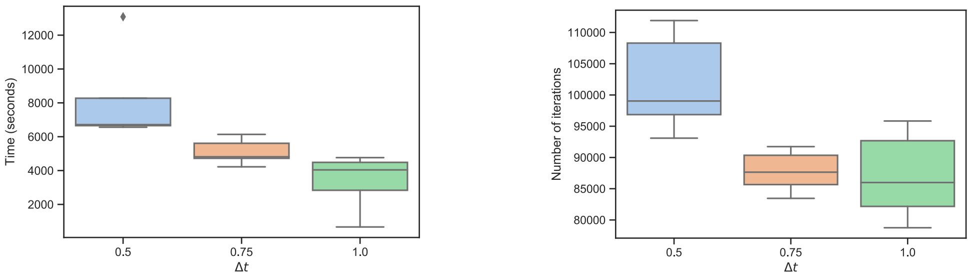


Figure 6.6: Computational efficiency of the variational approximation for each of the different methods. Results show the typical time taken for 100 iterations of the variational inference algorithms described in Chapter 5 for the MLFM-AG and the MLFM-SA method with order $M \in \{3, 5, 10\}$ on S^2 with a single latent force. Left: Time taken for 100 iterations when the MLFM is simulated on the interval $[0, 6]$. Right: time taken for 100 iterations when the MLFM is simulated on the interval $[0, 9]$.

100 iterates of the mean field variational inference algorithm for the MLFM-AG method and the MLFM-SA method with order $M \in \{3, 5, 10\}$. The results displayed in Figure 6.6 concur with the discussion above, with the MLFM-SA-3 method typically taking over three times longer than the MLFM-AG method on the length of interval, and then increasing at an approximately linear rate as the order increases after than.

Whether the increased time taken for the MLFM-SA method over the MLFM-AG is acceptable must be compared with the breakdown in relative performance already presented in Table 6.4, and the absolute performance as discussed in Section 6.2.5. When data is collected at a high sampling rate then the accuracy of the MLFM-AG method in this setting, combined with the superior computational efficiency would naturally suggest this method is to be preferred. However, in situations with much sparser data then accepting the increased computational burden of the MLFM-SA method with a high order may become unavoidable.

The above discussion quantifies the time taken for the variational methods. To complete our discussion we will also be interested in the time taken when using the Gibbs sampling approach to the MLFM-AG method as discussed in Chapter 3, compared to the variational methods introduced in Chapter 5. We consider the same experimental setup as in Section 6.3.3 for the case $T = 6$, that is we generated 100 datasets by simulating the MLFM on the interval $[0, 6]$ with a single latent force, we then sampled from each of these continuous trajectories with frequency $\Delta t \in \{0.5, 0.75, 1.0\}$. Convergence was monitored by use of the potential scale reduction factor (PSRF) [Gelman and Rubin, 1992]. A PSRF < 1.1 for values of the learned fiinite dimensional representation of the latent force was taken as an indication of sufficient convergence having been achieved. We used a Metropolis-Hastings scheme to sample from the joint density (3.46) as discussed in Section 3.4. Figure 6.7a displays the absolute time taken to achieve a sufficiently low PSRF to declare convergence. Since as the sampling frequency decreases the size of our dataset decreases then our discussion above of the theoretical complexity that the total time taken should decrease, and we observe results consistent with this in Figure 6.7a. Arguably of more relevance than the absolute time taken, is the total number of steps needed to achieve convergence. This is displayed in Figure 6.7b, while for this experiment we also observe the general decrease in the number of steps necessary to achieve convergence this is somewhat offset by the high variance observed for $\Delta t = 1.0$, this is likely caused by interpolant being poorly determined



(a) Execution time for 15000 MCMC steps

(b) Number of steps to convergence

Figure 6.7: Computational efficiency when using MCMC sampling for the MLFM-AG method on the interval $[0, 6]$ with varying sample frequencies $\Delta t \in \{0.5, 0.75, 1.\}$. (a) Time taken for 15×10^4 MCMC iterations. (b) Number of MCMC iterations to achieve convergence, (PSRF ≤ 1.1). The horizontal bars with each of the boxplots represent the median, the box margins show the 25th and 7th percentiles, finally the whiskers indicate data within twice the interquartile range, and the diamonds indicate outliers.

when there is a large temporal gap between successive observations, and in turn this leads to poor determination, and so convergence, of the values of the expected value of latent force.

6.4 Discussion

In this chapter we have presented a simulation study of the methods introduced in Chapter 3 and Chapter 4 of this thesis. Our investigation has been primarily concerned with the ability of these approximations to return reliable point, and distributional, estimates of the component functions in the coefficient matrix of the MLFM.

While for the most general cases of the MLFM it is not possible to obtain a reliable model of the true distributional structure the examination of the Kubo oscillator in Section 6.2.1 allowed us to compare the performance of our methods with an exactly solvable model. The results presented here have shown that for datasets with low temporal spacing between observations that the adaptive gradient matching method displays superior performance to the successive approximation method, with the additional benefit of possessing a simpler computational structure.

However, as the data becomes sparser the flexibility to tune the successive approximation method by increasing the truncation order leads to the potential for this method to demonstrate superior performance. This improved performance, in turn, is offset by an increase in the total interval size so that both methods perform poorly across long periods with large spaces between observations, a conclusion which is to be expected if one wishes to model long running time series with sparse data.

In an attempt to rectify this information loss over longer time intervals we considered a combination of local experts in Section 6.2.6. The results displayed in Figure 6.2 suggest that this method has the potential to alleviate some of the deteriorating performance of the MLFM-SA method over longer time intervals, and we advocate further research in this direction in Section 8.2.2.

For more complex geometries it is no longer straightforward to derive a suitable ground truth summary against which to assess our methods. In Section 6.3 we proposed three criteria by which we proposed to evaluate the methods we have introduced. The first of these was an approximation to how well the model recovers the parameters learned by minimising the parameters using least squares in the data space, and so acts as an approximation to the ground truth point estimate of the parameter. The results reported in Section 6.3.1 suggested that the MLFM-AG method performed very well at a high sample frequency, but deteriorates when applied to sparser data. Similarly, the MLFM-SA required increasingly high orders as the length of the interval increased. These conclusions were consistent with the results of the solvable model.

Our second criteria was a comparison of the mean field approximations obtained using the MLFM-AG method and the MLFM-SA method. Reassuringly the results presented in Section 6.3.2 are consistent with the observations from our previous experiments. In those regions where we would expect both methods to perform well against the unknown true distribution, they are also consistent with one another. When both methods are performing equally we would, all else being equal, prefer to use the MLFM-AG method. For the remaining entries in Table 6.4 there is a typically a substantial disagreement between our two methods of approximations, in this scenarios our previous experiments would best guide our choice of method. In particular if the sampling density is high we would prefer the MLFM-AG method; conversely, if the samples are relatively sparse, we would prefer the MLFM-SA method with a suitable high approximation order.

Our final assessment was applied only to the MLFM-AG method where we compared the sample moments of an MCMC sample from the posterior distribution with the results from the mean field factorisation. These results show that when we have access to a dense sample, the results are mostly equivalent, and so the increased efficiency of the deterministic variational methods makes them very attractive. For the other experimental settings the two approaches disagree substantially, but at the same time the totality of the results in this chapter suggest that in these regions the MLFM-AG method is not performing well, and this is true regardless of whether we are using MCMC or variational Bayes sampling.

In summary, the results of this chapter lend substantial evidence to the concerns we have raised about the information loss in the MLFM-AG on relatively sparse data, and it was this concern which motivated us to introduce the MLFM-SA method in Chapter 4. Our simulation studies suggest that the MLFM-SA method, by having an additional tunable order parameter, can eventually recover a good approximation to the posterior in regions where there is no current mechanism for the MLFM-AG method to retrieve this data. The studies in this chapter provide supporting evidence to justify using these methods in specific scenarios, but clearly, both methods require a deeper understanding of their theoretical properties, and we repeat this point in our final chapter.

Chapter 7

Application to human MOCAP data

7.1 Introduction

In this thesis, we have introduced a class of dynamic systems models for carrying out statistical inference for models with non-Gaussian trajectories. Many notable examples of such systems are realised either as the action of a rotation group or as a dynamic system on the rotation group itself. This is particularly true for complex, rigid structures that may be viewed as the composition of several segments of fixed length. The global position of such a body may then be represented by the position of a single point along with the relative orientations of the fixed length components. In this way, each segment may be viewed as an orientation in a local reference frame, and the motion of the whole collection can be described by the relative rotations of each orientation vector.

The human skeleton gives an important example of this sort of rigid system; the study of human motion data has been of significant interest in fields ranging from bio-mechanics and physiology to robotics and computer vision. Broadly speaking existing approaches to this problem can be categorised by the desired trade-off between constructing realistic mechanistic models of human motion versus the desire to obtain smooth, plausible animations of human motion, again this categorisation mirrors the mechanistic versus data-driven paradigm we have discussed in this thesis.

If a fully specified mechanistic model is desired then existing modelling approaches may include the specification of physically plausible models for the kinematics of human motion, perhaps with reference to toy physical control systems such as the inverted pendulum [Kuo et al., 2005, Kwon and Hodgins, 2017, An, 1984, Hall, 2015]. On the other hand, if the desire is only to attempt to fit smooth paths to existing data, perhaps for use in animation or prediction, then approaches based on smoothing and interpolation of more massive datasets may be used. A review of human motion with an emphasis on providing computer animation and fitting smooth paths to motion capture time courses is given in [Multon et al., 1999]

Between these two extremes, we have the LFM framework and [Alvarez et al., 2009] presents a successful application of the linear LFM to human motion data. In this section, we extend that work by using our MLFM to attempt to construct simple dynamic systems models which respect the geometric structure. In this chapter, our goal is to demonstrate the feasibility of the MLFM model for modelling the dynamics of data generated by observations. The fixed length of the human skeleton naturally suggests describing the motions of each segment as a relative rotation, and we aim to demonstrate the efficacy of the product manifold latent variable modelling approach we described in Section 2.3.2.

The process of representing the dynamics of these systems is referred to as (human) motion capture (mocap). The goal of motion capture systems is to measure a dynamical system of human body poses and to provide an effective representation of these motions allowing for further analysis. In Section 7.2 we discuss our data which is typically represented in as time series of rotation valued data. This allows us to consider modelling each segment as being an instance of the MLFM on the rotation group $SO(3)$, or a proper subgroup, and the whole skeleton as representing an instance of our product manifold construction.

We will demonstrate that it is possible to individually model each component of the skeleton using an instance of the MLFM with a distinct set of latent forces, but under the latent variable philosophy, it should be possible to construct an adequate description of the motion with a much smaller set of shared latent forces. In this chapter we shall show that this is indeed the case, we first carry out the modelling of the marginal trajectories for each joint independently in Section 7.3 for a fixed number of latent forces, and then in Section 7.4 we demonstrate the joint segment model fit using a shared set of latent forces. By way of cross-validation, we are able to demonstrate that the fit of the shared variables compares favourably with the independent fits. Therefore allowing for a much smaller representation of the complete dataset, and demonstrating the model's ability to capture information shared between the joints. We conclude with a discussion of the results presented in the chapter, comparing our approach with existing latent variable techniques for modelling human motion.

7.2 Motion capture data

Motion capture devices allow the recording of real-time motion by tracking the position of a collection of marker points over time. The time series of each marker may then be stored as a three-dimensional digital representation in some global coordinate system. The captured subject can be anything that exists in the real world with the critical points on the object positioned such that they best represent the orientations of the moving parts of the object, for example, the joint or pivot points in articulated objects. In order to accurately triangulate the marker positions more than one camera must be used with typically at least four required for accurate results.

Rather than record the position of the markers in a global coordinate system it has proved more useful to record this data in the form of relative positions. A representative 'skeleton' is constructed from a set of 'bone segments' of fixed length connecting the markers. The connection of these segments is given by specifying a hierarchical ordering such that each marker is connected to a single parent, apart from a designated root node. Each segment will then have a position in a local coordinate frame, the origin corresponding to the point of its parent in the hierarchy, as well as having coordinates in a global frame obtained by traversing the skeleton's hierarchy. Under this specification the motion can be described using only the local rotation of each bone segment and therefore any, not necessarily human, motion capture time series may be considered as a dynamic system on a collection of time-dependent rotations along with a global translation of the root node. In general the structure of the system will not allow arbitrary configurations of the connecting segments, and so most joints are also specified with a maximum allowable rotation (although we ignore this constraint) as well as the possibility of restricting the allowed rotations to a single 'degree of freedom channel', i.e. to only allow rotation around a subset of axes.

The dataset we use is from the Carnegie Mellon University (CMU) motion capture database which stores data in the ASF/AMC file format developed by video game company Acclaim. Two files describe a particular motion in the database; the Acclaim Skeleton File (ASF) which defines the base pose and skeleton hierarchy, and an Acclaim Motion Capture (AMC) file which describes the time series of motions. An example of part of a .asf file is displayed in Figure 7.1a. This file gives the base pose by specifying direction vectors relative to their parent as well as any of the constraints we have just discussed. Inspecting the line beginning 'dof,' we can read off the axes around which the *lfemur* segment is allowed to rotate; in this case it has the full three degrees of freedom. A second file, the .amc file, provides the data for each frame of motion capture. Part of a typical .amc file is displayed in Figure 7.1b where we see the data for frame one.

When we previously considered modelling rotation-valued data in Section 6.3 we considered the representation of a given rotation, $\mathcal{R}$, acting on $\mathbb{R}^3$ as being represented by a 3×3 real-valued matrix with orthonormal columns. Any such rotation matrix in $\mathbb{R}^{3\times 3}$ can be given an equivalent representation as the composition of three elementary rotation matrices, one around each axis. That is there is a representation of $\mathcal{R}$ in the form

$$\mathcal{R} = \mathcal{R}_y(\theta_y)\mathcal{R}_z(\theta_z)\mathcal{R}_x(\theta_x),$$

where $\mathcal{R}_x(\theta_x)$ represents a rotation around the x-axis of θ_x degrees, with similar interpretations for $\mathcal{R}_y$ and $\mathcal{R}_z$. These are referred to the Euler angle, or Tait-Bryan angles depending on the convention used [Whittaker and McCrae, 1947], representations of the rotation. As a three dimensional summary of the rotation group, they are convenient for storage, and it is these values that are reported in the .amc file. Because these matrices do not commute the order in which we take the combination of elementary rotation matters when attempting to reconstruct the global position of the skeleton. The ordering used for a motion represented by a particular .asf file is given on the line beginning with 'axis', for example in Figure 7.1a we see that the rotation is given by an initial rotation around the x-axis, followed by a rotation around the new y-axis, and finally a rotation around the z-axis.

For the experiments reported in this chapter we use motions $1-5$ from subject 64 in the CMU mocap database. Each of the motions records a single golf swing by the same subject. The data is recorded at 120 frames per second (fps) which we downsample to just over 6 fps. Because of the relatively high sampling frequency of the data, even after the downsampling step, we use the MLFM-AG method to carry out the modelling in this chapter.

7.3 Single segment modelling

Since each joint in our dataset can be represented as a rotation valued time series, and each joint represents the minimum unit for which we have prior geometric knowledge, we could consider modelling the whole skeleton by learning an instance of the MLFM independently for each segment in our dataset. This specification allows for no interactions between the joints, this is an unrealistic assumption, and we address the shortcomings of this assumption when considering the product manifold construction in the next section.

Our results in this section will demonstrate the estimates obtained when we fit each

```
:root
  order TX TY TZ RX RY RZ
  axis XYZ
  position 0 0 0
  orientation 0 0 0
:bonedata
  begin
      id 1
      name lhipjoint
      direction 0.62366 -0.717238
      length 2.53602
      axis 0 0 0 XYZ
  end
  begin
      id 2
      name lfemur
      direction 0.34202
      length 7.40817
      axis 0 0 20 XYZ
      dof rx ry rz
      limits (-160.0 20.)
             (-70.0 70.0)
             (-60.0 70.0)
end
```

(a) .asf file excerpt

```
1
root -5.82914 17.8741 1.78981 -78
lowerback 4.63582 1.57939 -1.4274
upperback 3.12177 2.18521 4.71147
thorax 0.382976 1.09183 5.58562
lowerneck -17.1013 -1.80218 2.337
upperneck 41.0205 0.947713 -5.860
head 18.3012 0.515299 -2.80985
rclavicle 6.7586e-015 3.8167e-014
rhumerus -4.17579 -3.52373 -100.9
rradius 19.9659
rwrist -31.1458
rhand -12.7873 -5.2607
rfingers 7.12502
rthumb 13.2982 -34.8164
lclavicle 6.7587e-015 3.8167e-014
lhumerus -11.0588 -43.7265 111.40
lradius 28.081
lwrist 45.5912
lhand -14.6248 69.8143
lfingers 7.12502
lthumb -168.473 80.5226
rfemur -29.9513 0.27183 12.9419
rtibia 29.1269
```

(b) .amc file excerpt

Figure 7.1: Examples of motion capture data recorded using the acclaim file format. The .asf file in Figure (a) records the hierarchical structure of the skeleton and specifies the data type of each channel, along with any degrees of freedom constraints. The width of both files has been cropped.

segment with $R = 2$ latent forces, this is convenient for visualisation purposes but will also be justified by our model selection analysis in the following section. We report the results for four different joints, the left/right humerus and the left/right femur. We chose a reduced set of joints to facilitate the presentation of our results; however, these joints still represent some of the most complex and variable motions for our golf swing data. Moreover, we shall see in examining this reduced set of segments that they possess a high signal-to-noise ratio allowing for interesting results and a compelling demonstration of our MLFM framework.

7.3.1 MLFM for quaternion valued data

As discussed in section 7.2 the data has been recorded as a time series of Euler angles, and as such we could consider mapping these to the corresponding rotation matrices and fitting a model similar to that considered in Section 6.3 on $SO(3)$. However, constructing a model in this way will lead to a nine-dimensional state variable. Rather than try to construct a model in the nine-dimensional space it will be more convenient to use the representation of $SO(3)$ as a subspace of the set of unit quaternions [Whittaker and McCrae, 1947].

Following [Moran, 1975, Prentice, 1987] a rotation matrix, R, may be identified with a unit quaternion $(q_0, q_1, q_2, q_3)^\top \in S^3 \subset \mathbb{R}^4$, where the coordinates are related by

$$R = \begin{pmatrix} q_0^2 + q_1^2 - p_2^2 - q_3^2 & 2(q_1 q_2 - q_0 q_3) & 2(q_0 q_2 + q_1 q_3) \\ 2(q_1 q_2 + q_0 q_3) & (p_0^2 + q_2^2 - q_1^2 - q_3^2) & 2(q_2 q_3 - q_0 q_1) \\ 2(q_1 q_3 - q_0 q_2) & 2(q_2 q_3 + q_0 q_1) & (q_0^2 + q_3^2 - q_1^2 - q_2^2) \end{pmatrix},$$

Instead of trying to model the trajectory on a group, we are now considering our data to be $\mathbb{R}^4$-valued and we wish to construct our model to leave S^3 invariant. Therefore we wish to consider the higher dimensional rotation group $SO(4)$ and model our observed data as having been generated by an action of this group on an initial unit quaternion. Compared to constructing the model directly on $SO(3)$ this leads to a higher dimensional Lie algebra, but a lower dimensional state variable and so we benefit from being able to avoid the numerical cost incurred by the need in the MLFM-AG method to place a GP prior on each dimension of the state variable.

The Lie algebra $\mathfrak{so}(4)$ of $SO(4)$ is given by the space of skew-symmetric 4×4 real-valued matrices with trace zero. This is a six-dimensional vector space. In the next section, we discuss the construction of the MLFM with the coefficient matrix supported on this vector space, and we chose our basis to be the canonical basis

$$\mathbf{L}_1 = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}, \qquad \mathbf{L}_2 = \begin{bmatrix} 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}, \qquad \mathbf{L}_3 = \begin{bmatrix} 0 & -1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix},$$

$$\mathbf{L}_4 = \begin{bmatrix} 0 & 0 & 0 & -1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix}, \qquad \mathbf{L}_5 = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 \\ 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}, \qquad \mathbf{L}_6 = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 \end{bmatrix}.$$

7.3.2 Model setup

In this section, we present the version of the MLFM which we will use to describe the motion of a particular bone segment. As discussed above we are focusing on only four segments from the upper and lower body, which will allow us a more thorough discussion of the results. For a given segment $seg \in \{lhumerus, rhumerus, lfemur, rfemur\}$ we specify the segment-specific model

$$\frac{\mathrm{d}\mathbf{x}^{seg}(t)}{\mathrm{d}t} = \left(\mathbf{A}_0^{seg} + \sum_{r=1}^{R} g_r^{seg}(t) \cdot \mathbf{A}_r^{seg} \right) \mathbf{x}^{seg}(t), \tag{7.1}$$

with

$$\mathbf{A}_r^{seg} = \sum_{d=1}^{6} \beta_{rd}^{seg} \cdot \mathbf{L}_d. \tag{7.2}$$

For each of the segments we place independent, identical prior distributions on the model parameters. For the connection coefficient parameters we assign the vague prior

$$\beta_{rd}^{seg} \sim \mathcal{N}(0, 10),$$

and these are assumed independent for $r = 0, 1, \ldots, R$, $d = 1, \ldots, 6$. Each of the latent force variables is assumed to have a Gaussian process prior with RBF kernel

$$\mathrm{Cov}\left\{ g_r^{seg}(t), g_{r'}^{seg}(t') \right\} = \delta_{rr'} \exp\left(-\frac{(t-t')^2}{2\psi_r^{seg}} \right),$$

where the length-scale parameter is given a generalised inverse Gaussian (GIG) distribution. This prior for the length scale parameters is recommended by the [Stan Development Team, 2018] to penalise both very large and very small length scales. This distribution is parameterised by three parameters, (a, b, p) [Johnson et al., 1994] and we write $\mathrm{GIG}(a, b, p)$. For this experiment we take the values

$$\psi_r^{seg} \sim \mathrm{GIG}(5, 5, -1),$$

and again these are independent for each $r = 1, \ldots, R$.

For the observation noise parameters we take relatively tight parameters

$$\sigma_k^{seg} \sim \mathcal{N}(0, 0.1),$$

independently for each $k = 1, \ldots, 4$. This choice is justified by the high precision and controlled environment of the mocap experiment.

Finally for the interpolating state variables we use a RBF kernel with both an absolute scale and a length scale parameter

$$\mathrm{Cov}\{x_k(t)x_k(t')\} = \phi_{0k}^{seg} \exp\left(-\frac{(t-t')}{2\phi_{1k}^{seg}} \right),$$

for $k = 1, \ldots, 4$.. We do not specify further priors for the hyperparameters $\boldsymbol{\phi}^{seg}$, in terms of the software we are implementing our model fitting on this is equivalent to the assumption of a vague uniform prior with large range on the log-transformed values of these variables. Note also that we do not include an absolute scale parameter for the latent force prior because the absolute value of these parameters is modulated by the coefficients β_{rd}^{seg}.

$$\mathrm{Cov}\{x_k(t)x_k(t')\} = \phi_{0k}^{seg} \exp\left(-\frac{(t-t')}{2\phi_{1k}^{seg}}\right),$$

for $k = 1, \ldots, 4$.. We do not specify further priors for the hyperparameters $\boldsymbol{\phi}^{seg}$, in terms of the software we are implementing our model fitting on this is equivalent to the assumption of a vague uniform prior with large range on the log-transformed values of these variables. Note also that we do not include an absolute scale parameter for the latent force prior because the absolute value of these parameters is modulated by the coefficients β_{rd}^{seg}.

7.3.3 Experimental results

We now consider the results obtained after applying the model just described to the golf swing data. For all of the experiments, our reported summary statistics are based on 1000 samples from the posterior distribution using Gibbs sampling with a burn-in of 10000 observations. To reduce sample autocorrelation, we recorded every 100th sample until the desired sample size had been achieved. The burn-in and total sample size was chosen so that for the reported experiments we achieved a PSRF of less than 1.2 as suggested in [Brooks and Gelman, 1998], this is of course problem dependent however our results appear relatively robust with similar conclusions drawn from an initial experiment with a substantially shorter Markov chain.

In Figure 7.2 we display the results obtained after fitting the model (7.1) to the *lhumerus* quaternion data. Figure 7.2a – 7.2d display the estimated predictive distribution of the trajectory in state space, along with the estimated ±2 standard deviation from the MCMC sample. Each output is of a similar absolute magnitude, and with largely similar frequencies of their variations. This conclusion is reflected in the MAP estimates of the kernel hyperparameters of the interpolating state processes which are

$$\begin{aligned}
\log \boldsymbol{\phi}_0^{lhumerus} &= (4.736, 4.593, 4.294, 4.180)^\top, \\
\log \boldsymbol{\phi}_1^{lhumerus} &= (0.081, 0.019, 0.089, -0.527)^\top.
\end{aligned}$$

The lower value of the length scale $\boldsymbol{\phi}_{14}^{lhumerus}$ is visible in a slight, low magnitude fluctuation during the early portion of the movement in Figure 7.2d, in comparison to the more regular motions for the remaining dimensions. While the effect is slight, it nevertheless does seem to be a genuine feature of this component of the trajectory, and this is indicative of both the importance and the challenges in currently tuning the hyperparameters of the state interpolating process.

Inspecting the learned latent force for the *lhumerus* data we observe an approximately sinusoidal force in Figure 7.2a reflecting the recurrence of positions in a golf swing, along with a second aperiodic force in Figure 7.2f reflecting a seeming change in the qualitative behaviour during the final part of the motion, roughly corresponding to the portion of the stroke after the ball strike.

The results for the *rhumerus* data are displayed in Figure 7.3. In comparison to the previous results, there is now a much greater degree of between motion variability, especially during the latter half of the motion and this is particularly evident in Figure 7.3c and Figure 7.3d. The greater irregularity of the *rhumerus* is further reflected in the latent forces displayed in Figure 7.3e and Figure 7.3f which display no obvious periodic behaviour, and seem to be more irregular than those observed for the left humerus. Taken together the greater variability of the predictive distribution and the increased irregularity of the latent forces would suggest inefficiencies in the right arm movement of this subject which would be worthy of further study.

The joints we have analysed so far are constituent parts of the left, and right arms, which go through an extensive range of motion during a golf swing, moreover smaller more complex joints are often hard to stabilise. On the other hand, lower body movement during a golf swing is much less dramatic. This can be observed in the more modest range of motions for the *lfemur* data in Figure 7.4, in particular, we note the scale of the right axis in Figure 7.4a. However, while the range of motion is relatively modest, there is still a relatively large degree of between motion variability. This, in turn, is likely to reduce the ability of the MLFM to learn an adequate representation of the dynamics, and we see that the latent forces displayed in Figure 7.4e and Figure 7.4f are much more regular over the full range of motion. Given that the left leg acts as a stable pivot point during a golf swing these results are mostly as we would expect.

Generally, for the four joints, we have examined in this section we observe more irregular forces for the left and right humerus segments than for the femur data. The greater complexity would seem to suggest that the controls required when performing a golf swing are more involved for the arm segments. These results fit with our intuition for the golf swing, with the comparatively larger leg joints being naturally stabilised during the pendulum motion of a golf swing. The upper body, and in particular the arms, which display a much higher degree of flexion during a given swing and so require more complex controls to achieve this motion.

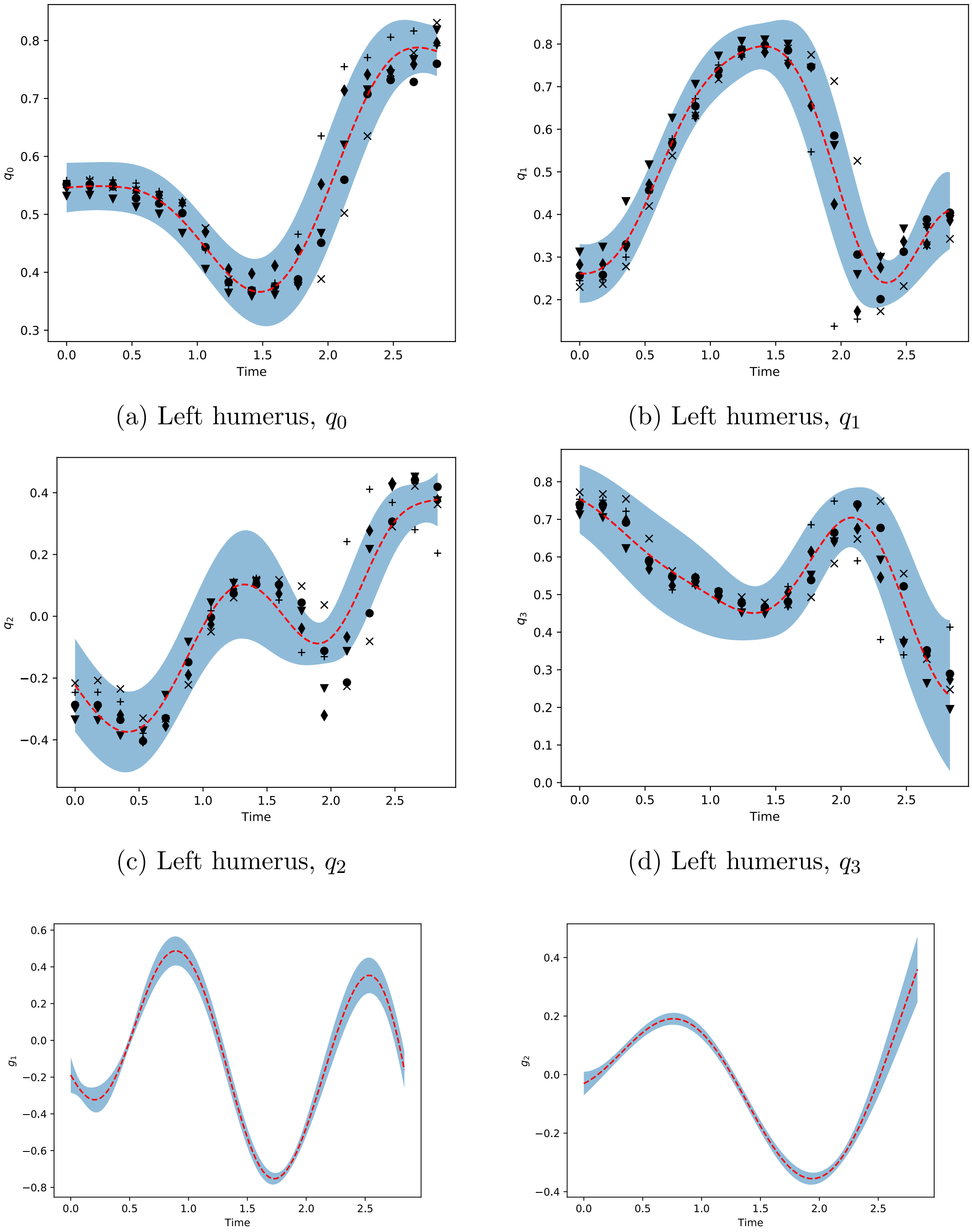
For the *rfemur* displayed in Figure 7.5 the predictive distributions seem very well determined. With these trajectories largely consistent between motions. Compared to the left femur data the latent forces have a more complex structure; in particular, we see an approximately sinusoidal control in Figure 7.5f.

To summarise the results of fitting the MLFM with two latent forces to the single segments has lead to models that have reasonably well informed predictive distributions. The most important aspect of the MLFM is learning the latent forcing functions from which the rest of the dynamic structure of a particular motion is recovered. In this section, we have seen that structure and regularity of the learned controls are consistent with our prior intuition about the relative complexity of each joint during a given golf swing suggesting that the model is accurately learning pertinent information. Indeed as a diagnostic tool, the model offers the potential to assess the relative efficiency of a particular motion by assessing the complexity of the latent forcing functions required to produce this motion. This process of discovery is ultimately what distinguishes the framework underlying the latent force model from optimal control theory.

Finally, we note that the learned forces seem significantly different between the segments, and it is not at all evident that any subset of these forces should be able to reconstruct this motion for the whole set accurately. Nevertheless, in the next section, we will show that it is, in fact, possible to jointly model the collection of bone segments with a common set of forces.

7.4 Joint modelling of multiple segments

In the previous section, we considered the possibility of modelling the rotation of a single joint of a motion capture human skeleton by identifying the rotation time series with a time series on the quaternion group. We were able to discover segment specific latent forces, and so learn dynamic systems which would recover the range of motion of individual joints during a golf swing. Our results in the previous section used a set of



(a) Left humerus, q_0

(b) Left humerus, q_1

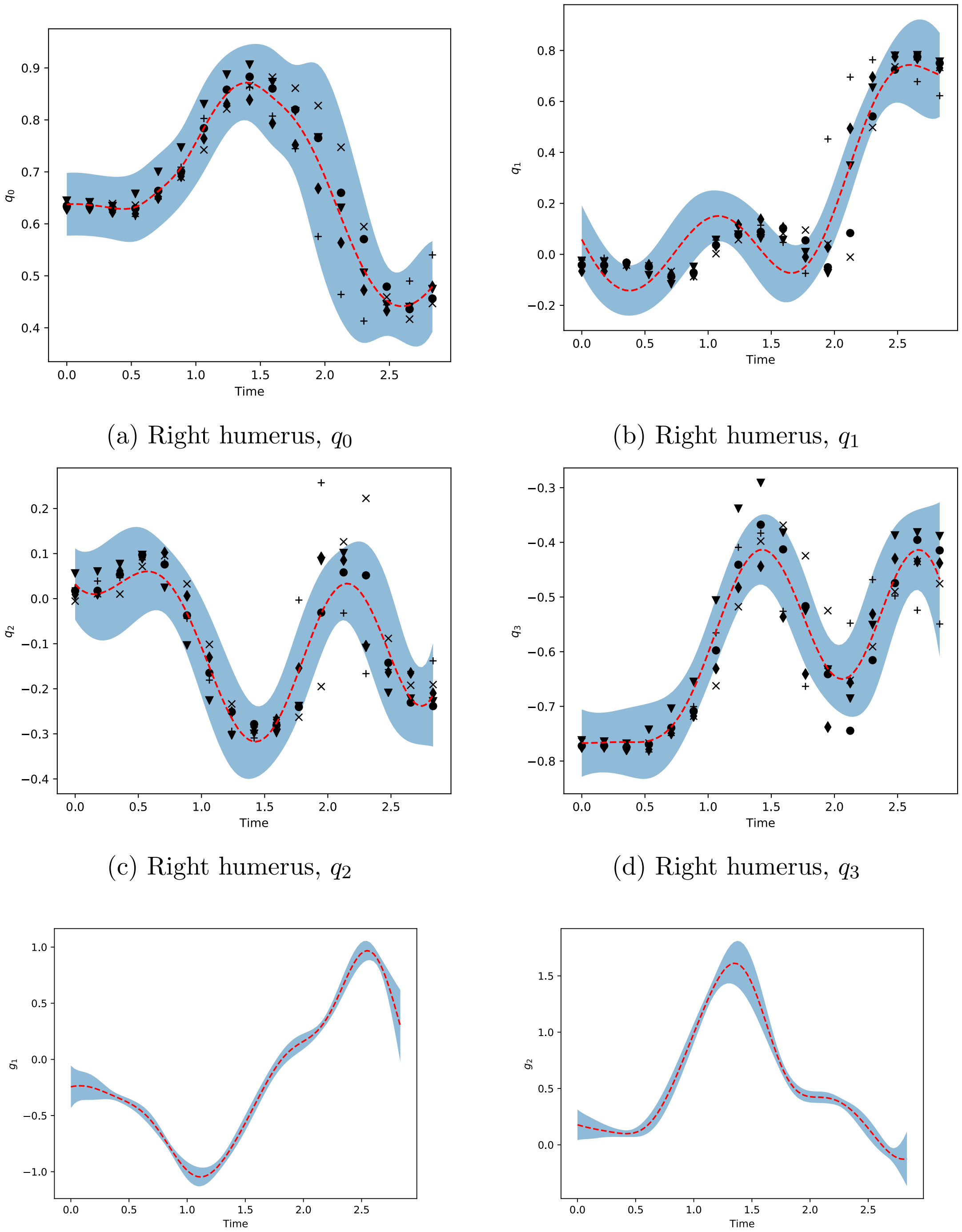
(c) Left humerus, q_2

(d) Left humerus, q_3

(e) Latent force 1

(f) Latent force 2

Figure 7.2: (a)–(d) Left humerus quaternion orientation for the set of motions 1-5 represented in increasing order by the symbols $\{\bullet, \blacktriangledown, \times, +, \}$. Also displayed are the marginal moments of the state variables represent by the mean value ‘- - -’, as well as the ± 2 standard derivation intervals, shaded region, based on 1000 samples of the state variables, latent force variables and the parameters β_{rd} from the posterior with the remaining parameter held fixed at their MAP estimates. The marginal moments of the latent force variables are displayed in Figures (e) – (f).



(a) Right humerus, q_0

(b) Right humerus, q_1

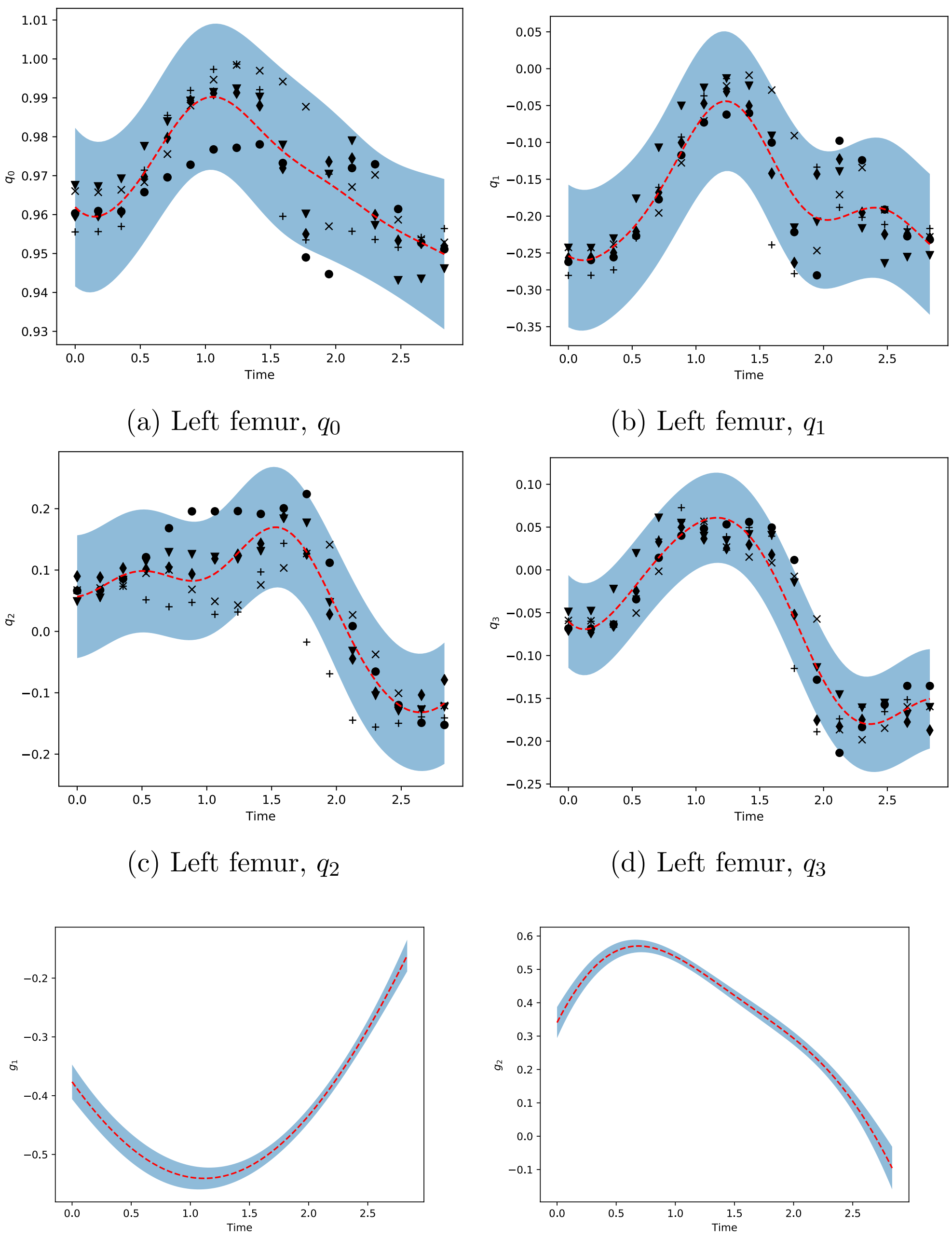
(c) Right humerus, q_2

(d) Right humerus, q_3

(e) Latent force 1

(f) Latent force 2

Figure 7.3: (a)–(d) Right humerus quaternion orientation for the set of motions 1-5 represented in increasing order by the symbols $\{\bullet, \blacktriangledown, \times, +, \}$. Also displayed are the marginal moments of the state variables represent by the mean value '- - -', as well as the ± 2 standard derivation intervals, shaded region, based on 1000 samples of the state variables, latent force variables and the parameters β_{rd} from the posterior with the remaining parameter held fixed at their MAP estimates. The marginal moments of the latent force variables are displayed in Figures (e) – (f).



(a) Left femur, q_0

(b) Left femur, q_1

(c) Left femur, q_2

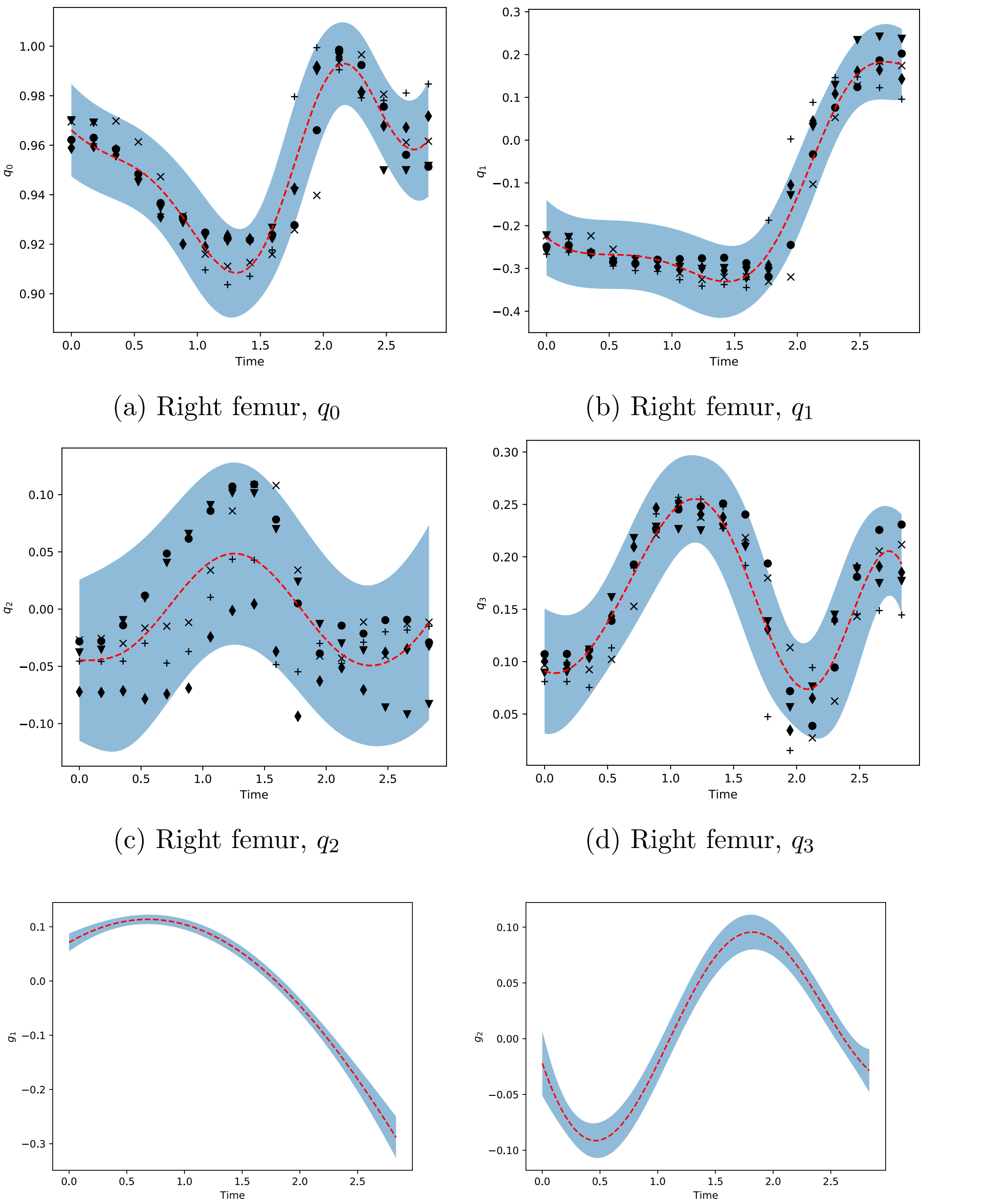
(d) Left femur, q_3

(e) Latent force 1

(f) Latent force 2

Figure 7.4: (a)–(d) Left femur quaternion orientation for the set of motions 1-5 represented in increasing order by the symbols $\{\bullet, \blacktriangledown, \times, +, \}$. Also displayed are the marginal moments of the state variables represent by the mean value '- - -', as well as the ± 2 standard derivation intervals, shaded region, based on 1000 samples of the state variables, latent force variables and the parameters β_{rd} from the posterior with the remaining parameter held fixed at their MAP estimates. The marginal moments of the latent force variables are displayed in Figures (e) – (f).

independent latent forces for each joint, and so each joint in the skeleton was treated as an independent dynamical system. The assumption of independence between segments for a particular motion is not only an unrealistic assumption but a restriction on the



(a) Right femur, q_0

(b) Right femur, q_1

(c) Right femur, q_2

(d) Right femur, q_3

(e) Latent force 1

(f) Latent force 2

Figure 7.5: (a)–(d) Right femur quaternion orientation for the set of motions 1-5 represented in increasing order by the symbols $\{\bullet, \blacktriangledown, \times, +, \}$. Also displayed are the marginal moments of the state variables represent by the mean value '- - -', as well as the ±2 standard derivation intervals, shaded region, based on 1000 samples of the state variables, latent force variables and the parameters β_{rd} from the posterior with the remaining parameter held fixed at their MAP estimates. The marginal moments of the latent force variables are displayed in Figures (e) – (f).

dimensionality reduction we can achieve, and so we would like to now couple each component of the dynamic system to allow information to be shared between segments and to achieve more substantial dimension reduction.

To achieve this goal we will make use of the product manifold version of the MLFM described in Section 2.3.2. With regards to the mathematical specification very little is changed with the new model taking the form

$$\frac{\mathrm{d}\mathbf{x}^{seg}(t)}{\mathrm{d}t} = \left(\mathbf{A}_0^{seg} + \sum_{r=1}^{R} g_r(t) \cdot \mathbf{A}_r^{seg} \right) \mathbf{x}^{seg}(t), \tag{7.3}$$

with

$$\mathbf{A}_r^{seg} = \sum_{d=1}^{6} \beta_{rd}^{seg} \cdot \mathbf{L}_d, \tag{7.4}$$

for $seg \in \{lhumerus, rhumerus, lfemur, rfemur\}$. Which is exactly the same as the single segment model (7.1) only the latent forces are now shared between segments. We shall use the same prior specification as described in Section 7.3.2. It remains to chose the total number of forces and in the next section we describe how to use cross-validation to make a principled choice of these forces, as well as justifying our decision to display the single segment fits with $R = 2$.

7.4.1 Experimental results and cross-validation

In the previous section, we presented the results for the single segment models with $R = 2$ forces. We now provide some more details about how this value was chosen for the single joints and compare the predictive performance of the single joint with results obtained using the multiple segment model. Since we have five replicates of the motion, we will determine the number of latent forces to be included by using leave-one-out cross validation [Hastie et al., 2009].

For each motion $m \in \{1, \dots, 5\}$ we train the model using all of those motions apart from the mth one. The predictions from the trained model are then compared with the actual observed outcome of the mth motion. We record the mean squared error of our prediction, and this is then averaged over the five possible configurations of the training and test sets. To make our prediction we use the sampling distribution obtained by MCMC sampling of the posterior, conditioned on the training data, for either the single segment or full collection models. As above we shall use a total of 1000 samples from the posterior distribution with a burn-in of 1000 observations.

The full results are displayed in Figure 7.6 which presents the point estimates for the mean squared error under both the single and complete segment models, as well as the plus-or-minus one standard deviation error bars for the single segment models only.

If we first interpret the results for the single segment models, then as a general comment we observe that in terms of the scale of error the values for the left and right humerus are more extensive than those for the femur segments. This is in line with our analysis of the plots in the previous section, as well as a reflection of the relative complexity of the arm motion during a golf swing compared to the lower body movement.

For all of the fits obtained, we observe that the model with only a single latent force displays the worst performance, and in all cases increasing the number of forces to $R = 2$ leads to an improvement. Beyond that, we may, in fact, experience an increase in the prediction error. This is likely due to the increase number of forces leading to overfitting of the model, with the latent forces accounting for dynamics which are better potentially better explained by state interactions and leading to a reduced predictive power of the model. This is most readily observed for the relatively more complicated left humerus data in Figure 7.6a and right humerus in Figure 7.6b. In general, the point estimate obtained for $R = 2$ in the single joint models is within one standard deviation of the minimum error, and this motivated our choice to display these results in the previous section.

With regards to the marginal improvement offered by increasing the number of latent forces from a single force to two forces we notice that the most significant benefits are observed for the left humerus, Figure 7.6a, and the right femur, Figure 7.6d. Recalling our analysis from Section 7.3.3 these were the forces which displayed both a single higher frequency sinusoidal force and a slower varying force. Being able to account for both of these dynamics lends significant predictive power to the learned dynamics, and it would seem that a single force is insufficient to capture both of these features of the driving forces. Our comments in the previous section regarding the complexity of the learned forces are evident in the number of forces needed to allow for accurate inferences, this is particularly evident for the left femur 7.6c, which as we noted when analysis Figure 7.4 has relatively uninteresting latent forces when $R = 2$, and from the cross-validation results we see a single latent force could equally well model the dynamics of this segment.

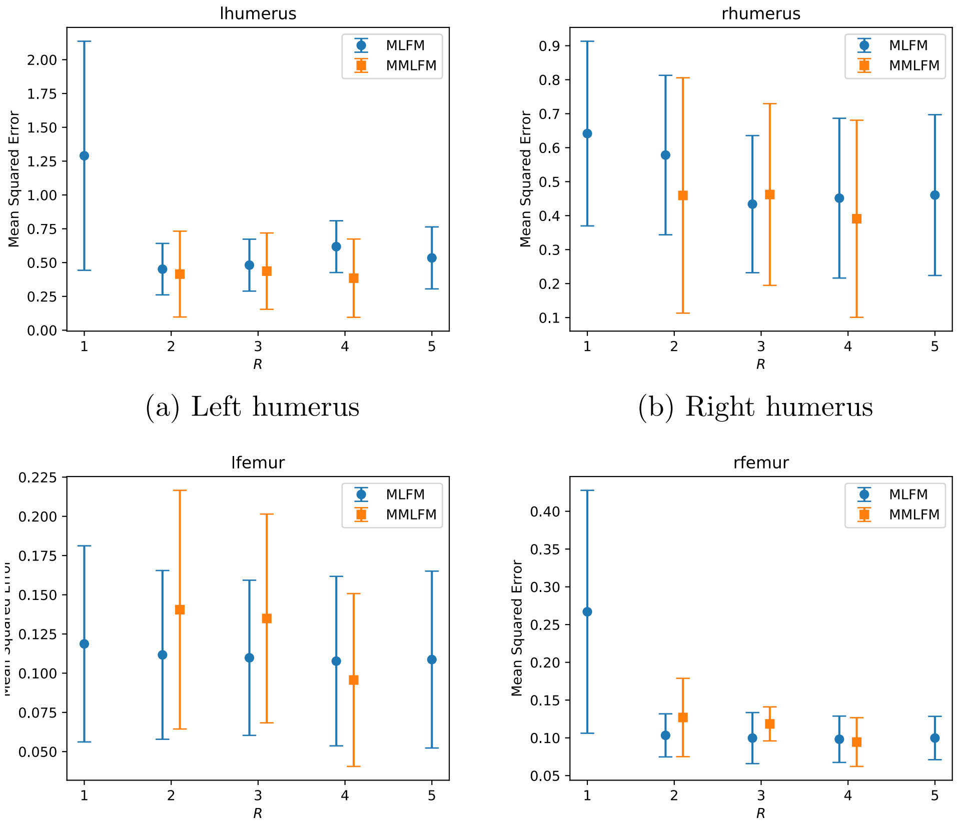
Learned latent forces in the full model

To further analyse the model, and add some context to the increasing predictive performance observed in Figure 7.6 as we increase the number of latent forces we plot, and comment on, the forces learned during the full segment fits.

We begin by considering the latent forces learned when $R = 2$, the results are displayed in Figure 7.7. The first latent force in Figure 7.7a displays relatively regular motion, with a more irregular force in Figure 7.7b. Both the learned forces have a turning point at roughly the two second mark, in this region the first force is close to zero so that the final phase of the golf swing dynamics in the $R = 2$ model are largely accounted for the by the second force, which displays a steep trend over this interval.

To carry out further analysis we now compare the qualitative properties of these forces to those obtained when we increase the number of latent GPs to $R = 4$. The results for the complete segment model with four forces is displayed in Figure 7.8. The forces recovered in 7.8b and 7.8c display fairly regular motions, with the magnitude of their variations largely consistent along the whole duration of the motion. We recall that all of the latent forces were assigned infinitely smooth, stationary priors and it would seem that on the passage to the posterior for these particular forces we preserve that qualitative behaviour. The third force, Figure 7.8c, has a turning point at roughly the halfway point of the interval, and at this value is close to zero, suggesting the input of this force reverses around the halfway mark, coinciding with the end of the back-swing and the beginning for the forward swing.

The fourth force displayed in Figure 7.8d, and certainly the force displayed in 7.8a seem qualitatively different to the other two forces. Both of these forces display relatively quiescent behaviour with slow oscillations for the majority of the interval, before dramatic final motions with almost constant velocity. Again this first period of the motion coincides with the controlled back-lift of the swing, and the second with the forward motion. This more dramatic range of final motion is also observed for the tightening of the posterior distribution for the latent force in 7.8b, even if not as directly obvious from the magnitude of the motion.



(a) Left humerus

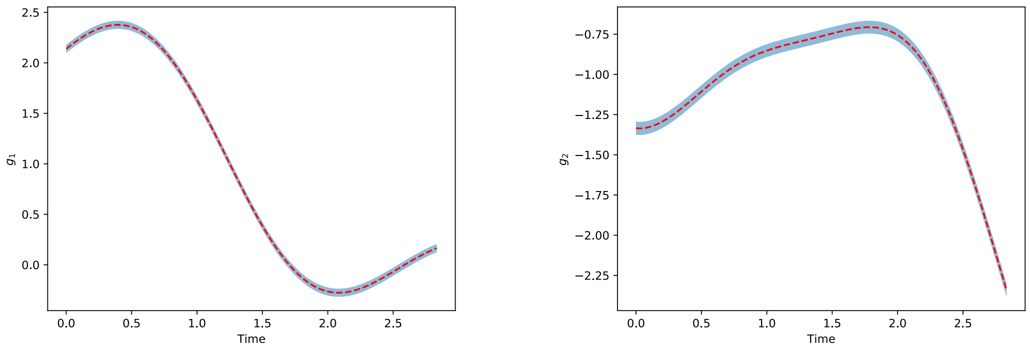
(b) Right humerus

(c) Left femur

(d) Right femur

Figure 7.6: Estimates of the prediction error obtained using leave-one-out cross validation with R latent forces for the asf bone segment. The error bars represent the estimated mean squared error for the single segment models along with plus-or-minus one standard deviation limits. We also display each of the joint prediction error for the Multiple manifold decomposition of the MLFM (MMLFM) with $R \in \{2, 3, 4\}$ latent forces when each of the joint share a common set of latent forces.

Comparing the two models we note that while the results in Figure 7.6 demonstrate that the model with four latent forces out performs that with only two, the improvements are still relatively modest in magnitude. And remarkably the results for $R = 2$ in the full segment fit were within one standard deviation of the best results obtained for the marginal moments. Suggesting not only that there is a high degree of information shared between the segments, but that the latent forces are able to capture this information sharing. In general the dynamics seem to be well accounted for by a slow varying force during the back and forward swing, with an additional degree of freedom needed to account for the final phase following the ball strike, and these components are captured by only two latent forces. Increasing the number of forces to four does lead to improved prediction, but not necessarily any increase in interpretability or understanding of the dynamics. Instead the model with more forces allows for effects already observed in the more parsimonious model to be further deconstructed, for example this specification has the luxury of specifying a force in Figure 7.8a that is almost entirely dedicated to controlling the final phase of motion.



(a) Latent Force 1

(b) Latent Force 2

Figure 7.7: The two latent forces learned using the product manifold MLFM for the complete set of segments with $R = 2$. The estimated mean value of these forces obtained using 1000 samples of the posterior using MCMC sampling is given by the '- - -' line. The shaded region represents the ±2 standard deviation intervals around the mean for the marginal moments of the sample

7.4.2 Comparison with the LFM

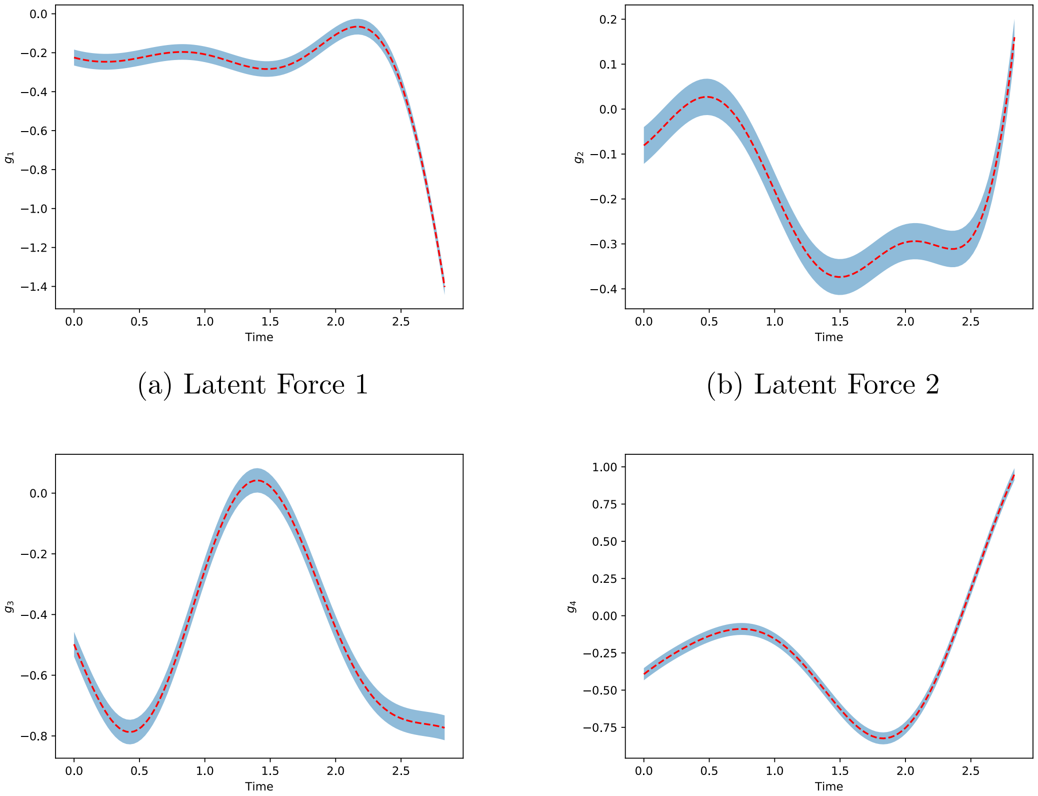
As a final examination of the model we compare the performance of the MLFM with the first order LFM described in Section 2.2. We use the same cross-validation procedure described above. That is we train our LFM and MLFM model on the complete data for four of the motions and use this to predict the outcome of the final motion. We let $q_k^{seg}(t)$ denote the kth component of the quaternion representation of the data for a given segment, then the LFM specifies a time-evolution equation

$$\frac{\mathrm{d}q_k^{seg}(t)}{\mathrm{d}t} = D_{\iota(seg,k)} \cdot q_k^{seg}(t) + \sum_{r=1}^{R} S_{\iota(seg,k)r} g_r(t), \tag{7.5}$$

for $k = 1, \ldots, 4$ and $seg \in \{lfemur, rfemur, lhumerus, rhumerus\}$. The sensitivity matrix is of shape $16 \times R$ and we denote by $\iota(seg,k) \in \{1, \ldots 16\}$ an enumeration of the segment and dimensional component. We use the same prior for the latent forces on the LFM, and the choice of RBF kernel allows us to use the closed form expression for the covariance given in [Alvarez et al., 2009].

The results are displayed in Table 7.1 and we see that for all values of the latent force variable the MLFM outperforms the LFM. Strongly suggesting that for complex, nonlinear time series such as those occuring for human motion data we can gain increased predictive performance by specifying a model is able to encode our prior knowledge of the geometric constraints, and to allow for nontrivial, but still linear, combinations of the state variable in the time-evolution equation rather than the restrictive diagonal structure of the LFM.

Given the relative complexity of human motion it is not surprising that the first order model (7.5) failed to appropriate describe the motion, and arguably a fairer test would include the second-order model considered in [Alvarez et al., 2009]. However, to our knowledge there is no readily available software for fitting this model and the second-order model is significantly more complicated to implement than the first-order model. Nevertheless, we do agree that a comparison with the second-order model would



(a) Latent Force 1

(b) Latent Force 2

(c) Latent Force 3

(d) Latent Force 4

Figure 7.8: The four latent forces learned using the product manifold MLFM for the complete set of segments with $R = 4$. The estimated mean value of these forces obtained using 1000 samples of the posterior using MCMC sampling is given by the '- - -' line. The shaded region represents the ± 2 standard deviation intervals around the mean for the marginal moments of the sample

be an important area for future study.

	R		
	2	3	4
LFM	0.699 ± 0.411	0.441 ± 0.392	0.337 ± 0.301
MLFM	0.285 ± 0.198	0.288 ± 0.160	0.241 ± 0.167

Table 7.1: Comparison of the mean squared prediction error for the LFM and MLFM using leave-one-out cross-validation on the complete set of segments. We report the point estimated of the prediction error along with the estimated standard deviation

7.5 Discussion

In this chapter, we have carried out the process of fitting the MLFM to time series of rotation valued data describing a complex human motion. There are at least three essential advantages our approach has over previous approaches to fitting smooth paths to rotation valued data as in [Prentice, 1987]. First, our methods do not require the use of a local linear approximation to the manifold. Secondly we do not just fit a smooth path, but also a dynamic system, and finally, we can jointly model a collection of rotation valued time series and so model a large complex system by a smaller number of free forcing functions.

Because our methods learn dynamic systems, and in particular ODEs, they allow for the straightforward generation of new motions, including the simple process of speeding up/slowing down these motions by using time changes of the driving forces. Furthermore, because the dynamics are driven by a small number of latent forces they provide an easy to interpret representation of the learned motions compared to data-driven approaches such as the use of recurrent neural networks in [Du et al., 2015, Fragkiadaki et al., 2015]. We have demonstrated this interpretive step in our analysis of the learned latent forces for the single segment model in Section 7.3, and the joint model in Section 7.4. In both cases, we were able to plot and examine qualitative features of the learned forces, something that would be entirely out of reach using alternative approaches such as recurrent neural networks in [Du et al., 2015, Fragkiadaki et al., 2015].

Our results have demonstrated the effectiveness of combining a small number of latent forces with a natural decomposition of the data space, in particular, our estimates of the prediction error in Figure 7.6 demonstrate that there is successful information sharing between the different manifold-valued processes using our MLFM framework. Alternative latent variable modelling approaches to human motion data have often tried to first achieve a more significant dimensionality reduction of the data space. Early approaches considered the use of PCA [Ormoneit et al., 2000, Sidenbladh et al., 2000, Yacoob and Black, 1998] assuming that the latent variable model could be well expressed by the projection of the data on to a lower dimensional linear subspace. Unfortunately, there are good reasons to believe that the latent variable manifold for human motion is nonlinear [Bissacco, 2005], and this can be observed in the improved performance of our geometrically constrained MLFM compared to the linear LFM discussed in Section 7.4.2. However, because our method involves a decomposition of the data space, we do not achieve the same scale of dimensionality reduction as alternative methods in the machine learning literature. In the next chapter, we begin to consider how best to combine our approach with some of these alternative methods to achieve both the rich topological structure we can exhibit with the MLFM and a more significant dimension reduction.

Motion capture systems such as the dataset we have been considering in this chapter have proved very useful for those involved in creating live-action cinema, but they have proved less popular for animated features. Typically animation requires a much more stylised representation of motion than that observed in real-world movements, and it is not necessarily intuitive for animators to jointly alter high dimensional time series of rotation valued data to realise a “stylized” transformation of natural motions. We believe the latent force modelling approach allows for a much more intuitive process of realising new transformations by allowing animators to manipulate the small number of latent forces we have learned in order to realise original motions.

While we have successfully demonstrated the potential of the hybrid modelling approach, we have only embedded geometric constraints into the model, and so there is still scope to add more physical realism to the type of models we are considering. For instance, the biological efficiency of the human body, and the natural stabilisation and spring-like properties of joint and limb segments should be viewed as having a mechanistic structure that complements the exogenous, guided controls that we have focused on modelling. Work remains to be done on adding further physical realism into hybrid models that allow these two important features to be included in such a way that they best complement one another.

Chapter 8

Discussion and future work

Our principal aim in this thesis has been on introducing a flexible class of models for performing predictive inference in complex dynamic systems without having to specify complete mechanistic models, and we have provided two methods of approximating the conditional distributions of this class of models. In this chapter, we first present a discussion of the work presented in this thesis and discuss several important directions for future work.

8.1 Review

8.1.1 A latent force model with multiplicative interactions

In this thesis, we have extended the latent force modelling framework introduced by [Lawrence et al., 2006, Alvarez et al., 2009] to allow for multiplicative interactions between the latent control variables and the state variables. While initially introduced as a mechanistic model it was quickly realised that such models were an effective way of embedding dynamic systems properties into a GP model by way of a nonstationary kernel function. This procedure allowed for the construction of hybrid methods combining an overly simplistic mechanistic model with the flexibility of GP regression methods. The ability to construct a dynamical systems model, but remain within the GP regression setting readily allows for existing inferential techniques used in the general regression setting to be immediately applied to the LFM, or for the LFM to be used in more complex models constructed around GPs.

While the fact that the LFM is a special case of the GP regression model leads to the analytic tractability of this model, it is also indicative of one of the most restrictive assumptions, namely the assumption of Gaussian trajectories. Directly related to the Gaussian trajectories property is the fact that such trajectories are necessarily supported on a vector space. A desire to move beyond this restrictive assumption has been a primary motivation in our introduction of the MLFM in this thesis, and so allowing for the modelling of trajectories with a richer geometric structure. At the same time we have endeavoured to remain faithful to the ingredients that allowed for the tractability of the LFM; that is to say a linear ODE for which the time dependent behaviour of the evolution equation arises from the fluctuations of a set of independent GPs.

The geometric constraints of the MLFM are achieved by constraining the time-dependent coefficient matrix to be an element of the Lie algebra associated with some matrix Lie group for all t in some interval. This is achieved by restricting the coefficient matrices to linear combinations of a set of basis matrices of a particular Lie algebra. This is sufficient, [Iserles, 2004], to ensure that the fundamental solution of the ODE is then an element of the matrix Lie group, and from this, it is then possible to either construct models constrained to the matrix Lie group or formed by the action of this group on a vector space.

An important feature of our MLFM is the ease with which we may build models with a rich topological structure. Such structure frequently appears in nature, for example, many physical processes are bounded and display periodic, or quasi-periodic motions, and this suggests that much of the dynamics of this motion are well explained by trajectories taking the form of quasi-periodic orbits on a compact manifold. The framework introduced in this thesis makes the process of constructing such a latent space straightforward; exactly periodic motion on a compact manifold can be achieved by the Kubo oscillator discussed in Section 6.2.1. The model can then be made more flexible, while still ensuring that the motions are compact, by considering the larger space S^2, which contains the Kubo oscillator S^1 as a nested submodel.

The necessity of considering the topology of the latent space has been well appreciated in the machine learning literature. The GP latent variable model with back constraints (BC-GP-LVM) [Lawrence and Quiñonero Candela, 2006] is an attempt to combine the flexibility of GP methods of dimensionality reduction with the use of back constraints to preserve local distances in the dimensionality reduction process. In applying the BC-GP-LVM to periodic human motion data [Lawrence and Quiñonero Candela, 2006] conclude that a circular structure will be necessary, but that a two-dimensional latent space is overly constrained and so unable to demonstrate the required structure adequately. Their solution is to increase the dimension of the latent variable space, but they also remark that the introduction of a cylindrical topology is also likely to resolve the issue. This conjecture is supported in [Urtasun et al., 2008] where the use of such a topology is demonstrated, and so it is instructive to consider how we might impose a cylinder topology using our MLFM framework. One possibility, using a single latent force, would be to consider a three dimensional state variable $\mathbf{x}(t) = \begin{bmatrix} x(t) & y(t) & z(t) \end{bmatrix}^{\top}$ governed by the differential equations

$$\frac{\mathrm{d}}{\mathrm{d}t}\begin{bmatrix} x(t) \\ y(t) \end{bmatrix} = (\beta_{01}^{(1)} + \beta_{11}^{(1)} g_1(t)) \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} x(t) \\ y(t) \end{bmatrix},$$

and

$$\frac{\mathrm{d}z(t)}{dt} = \beta_{01}^{(2)} z(t) + \beta_{11}^{(2)} g_1(t),$$

which is an example of the Kubo oscillator model on S^1, and a univariate LFM on $\mathbb{R}$, combined using the product manifold construction from Section 2.3.2. This model then has the flexibility to wrap around the cylinder $S^1 \times \mathbb{R}$, with the non-autonomous dynamics allowed by the GP control functions allowing the trajectories to self-intersect in phase space giving the model more expressive flexibility, while the topological construction ensures that the trajectories are constrained to a cylinder. By moving beyond the GP model, we are free from the constraint of embedding topological properties entirely through the choice of the kernel as would be necessary for the GP setting. This avoids the problem of trying to construct the topological property invariance through the choice of the kernel itself, which in general is not easy, and naturally suggests ways of building more complex models. Indeed the construction in [Urtasun et al., 2008] is possible within the GP regression framework because of the relative ease with which we can construct periodic kernels, but it is not clear how to construct kernels which could reflect more general topological features, note for instance that the cylinder $S^1 \times \mathbb{R}^1$ is not compact unlike the topology of S^2 discussed above.

Of course what our model lacks relative to the BC-GP-LVM, and similar methods of dimensionality reduction is an operation to map a higher dimensional data space to this cylinder, and more generally our model achieves only a modest dimensionality reduction because of our focus on embedding known geometric constraints. Instead, our model reduction is better viewed as the ability to construct flexible and parsimonious evolution equations in the data space, the parsimony being achieved by the number of latent forces R. It would be desirable to combine this more directly with a dimension reduction, and we discuss the addition of this feature later on in this chapter when we discuss future work.

While we do not achieve a dimension reduction in the data space an important aspect of the product manifold modelling approach as described in Section 2.3.2 is the ability to jointly model multiple systems each constrained to some submanifold, and therefore we are able to achieve a great deal of control, and ability to encode relevant prior knowledge, when specifying this decomposition. We demonstrate the potential of this system for predictive inference in dynamic systems in Chapter 7 where we show that the joint modelling of several connected joints leads to superior predictive performance over the marginal joining with only a modest increase in the number of latent forces required.

In summary, the latent force model with multiplicative interactions between the state and controlling GP variables that we have introduced in this thesis enables the embedding of geometric constraints into models of dynamic systems. Our construction allows the specification of models with a higher degree of physical realism while still being flexible enough to be broadly applicable, even in those cases where a complete system specification is hard to motivate. Unfortunately, the ability to model trajectories on non-linear manifolds comes at the expense of the tractable inference of the LFM. This arises because the mapping from the latent Gaussian processes is no longer a linear map, and in general is given by the time ordered integrals discussed in Section 2.3.1. Because of the intractability of the transformation, we need to consider methods of approximate inference, and we now review the two methods introduced in this thesis.

8.1.2 Adaptive gradient matching

Our first attempt at deriving an approximate probability density function for the MLFM used the Bayesian adaptive gradient matching methods developed in [Calderhead et al., 2009, Dondelinger et al., 2013]. The simple observation motivating this construction is that the time-evolution equation provides an explicit, and in the case of the MLFM linear, relationship between the parameters of the model and the trajectory through its gradient, so that inference can be guided by the combination of parameters and the latent states that 'match' the gradient. In practice the gradient is unobserved, and so the method works by completing the model with the gradient using an approximation to the conditional distribution $p(\dot{\mathbf{X}} \mid \mathbf{X}, \boldsymbol{\theta})$ and then marginalising out the gradient.

In general the adaptive gradient matching method is only able to produce a conditional density up to an unknown normalising constant, however, for every variable appearing linearly in the time-evolution equation, it is possible to rearrange this unnormalised density as an exponential quadratic in this variable. As an immediate result for all those variables which have a Gaussian prior, we can realise a Gaussian posterior conditional distribution. For a particular variable, the conditioning must be done on all those variables which interact multiplicatively with it in the time-evolution equation, along with the hyperparameters of the latent state interpolants and the regularisation parameter of the gradient expert. In Section 3.3.1 we carried out this process and presented the Gaussian posteriors for the latent states, latent forces and connection coefficients.

The existence of tractable posteriors for an important subset of the model parameters makes it straightforward to carry out a Gibbs sampling routine using the tractable conditional distributions. This same structure also lends itself well to mean field variational approximations of the posterior distribution, and we presented these in Section 5.3.1. We also discussed MAP estimate of the model parameters noting that because it is possible to marginalise out the state variable, there is little to be gained using the EM algorithm for MAP estimation.

The approximation to the conditional distribution $p(\dot{\mathbf{X}} \mid \mathbf{X}, \boldsymbol{\theta})$ introduced in [Calderhead et al., 2009] takes the form of a multiplicative product of experts. The first expert corresponds to a smooth GP prior on the state and its gradient, and the second takes the form of a regression model introducing a regularisation parameter controlling how strictly the functional constraint implied by the time-evolution equation is imposed. While intuitively appealing and demonstrated in Chapter 6 to work well in scenarios with a high frequency of sampling there are as yet no rigorous guarantees for the accuracy of this approximation. A justifiable concern is the performance of the method as the distance between samples increases, and therefore the likelihood of the interpolating gradient properly belonging to the tangent space of the true manifold. Graphically this deteriorating performance is indicated in Figure 6.1 and the conjectured deterioration in performance borne out by our simulation studies.

In the finite dimensional case, some work has been done establishing the consistency of parameter inference when the estimator of the trajectory is given using a kernel density estimator. For the setting we are interested in it should be relatively straightforward to adopt such methods to the case where the plug-in estimator for the trajectory is a Gaussian process. However, the dependence of the results on the bandwidth or more general properties of the trajectory is still poorly understood, and this will also be the case for the role of the hyperparameters of the GP interpolant. Establishing more rigorous guarantees of this method is an important topic of future research, and we discuss this further in Section 8.2.1.

Despite the caveats just mentioned when we have access to data observed with sufficient frequency the MLFM-AG method provides an effective and efficient means of constructing point estimates and distributional estimates for the MLFM. This also makes it an attractive candidate for approximate inference in more complex models which feature the MLFM as a particular subsystem, for example, the latent space manifold exploration problem which we shall discuss in Section 8.2.3.

8.1.3 Method of successive approximations

By introducing an interpolant of the trajectory the MLFM-AG method avoided the need to ever explicitly solve the ODE, and instead, it was possible to introduce an approximation to the trajectory. As we have discussed in Section 6.2.4 the main drawback is the potentially limited ability of gradient matching approaches to accurately recover structural properties of the trajectory away from the training points, although as discussed in Chapter 3 the adaptive gradient matching method of [Dondelinger et al., 2013] does allow for local parameter dependent regularisation of the interpolants. Methods based on solving the ODE do not have this problem because each realisation of the solution is a valid trajectory from this model, conditional on the parameters, and so accurately recovers the structural information. This has motivated our attempt in Chapter 4 to introduce a method which is more faithful to the idea of solving the ODE, but avoids the propagation of the latent variables through the solution of a numerical ODE.

The first step was to rewrite the ODE as an equivalent integral equation. We could then approximate this integral using some numerical quadrature rule and so for a linear ODE, such the MLFM, realise a system of linear equations. The result is a set of linear constraints which must be satisfied by the model parameters, and so after rearrangement leads to a method of estimating the parameters as the solution of a system of linear equations. However, as we discussed at the start of Section 4.3, the construction of an accurate quadrature rule will require completion of the trajectory. In the absence of a suitable approximation to the trajectory it is not clear how this completion can be achieved and leads to a similar problem as experience in the MLFM-AG method; how do we approximate the distribution of the trajectory away from the observed time points.

The problems we have just discussed arise because we are attempting to approximate the marginal distribution of a trajectory which is given only implicitly. Therefore, rather than attempting to approximate this marginal distribution, we consider using a generative method to build an approximation using explicit transformations of a quantity with known initial distribution. A key component in our construction is the method of successive approximations; a standard tool in the classical existence and uniqueness theorems for ODEs. This allowed for the introduction of a truncated series expansion of the trajectory, with each term in the expansion given as an integral transformation of the preceding term.

In general, it will be necessary to consider the completion of each of the successive approximations to allow for accurate quadrature. However, since each term in the expansion is an explicit transform of the preceding term, and the distribution of the first term is known for arbitrarily fine realisations, we do not experience the same problem when approximating the distribution of the completion. Having generated a set of successive approximations with known distribution we are now back in a situation where we can utilise the implied linear constraints to estimate the parameters, only now we will have multiple linear systems with the solution given by the intersection of these hyperplanes.

Of course, the immediate downside of this method is the vast increase in the variable set by retaining the complete set of approximations, one option is to marginalise over this larger variable set, but the resulting likelihood term for the trajectory conditional on the parameters has a complex nonlinear dependence on the parameters. As an alternative we discussed in Chapter 5 we discussed the possibility of using variational methods that combine the ability to exploit the conditional structure of the full model, with a marginalisation step to allow for efficiency. The linear dynamical systems structure of the successive approximations allowed the use of Kalman filtering so that we avoided the need ever to retain the joint distribution of the successive approximations, and instead only need to store statistics involving the pairwise expectations during each step.

A further downside of this method, and one which is not improved by the use of variational methods, is the fact that this construction is local, with the accuracy decaying as we move away from an initial condition. To attempt to alleviate this problem we have considered the possibility of combining several local approximations of a lower order through a mixture modelling approach. Our simulation studies in Section 6.2.6 demonstrate that increasing the number of mixture components can lead to a successful approximation of the density by a mixture of lower order approximations.

However, there are caveats to the mixture modelling approach. Crucially this approach will reintroduce a bandwidth type parameter controlling how we combine these mixtures, although potentially at a level removed from the use of a kernel density estimator directly on the trajectory as used in the collocation/smoothing approaches. The Gaussian mixture modelling approaching leads to a 'soft assignment' of the variables to their respective classes, and so requires the evaluation of each mixture density for the whole sample, this is potentially computationally efficient and also numerically unstable for points far away from a given initial condition due to the polynomial structure of the approximation. In discussing the connection between the deterministic form the MLFM-SA and feed-forward neural networks with linear activation functions we discussed the possibility of using temporal convolution layers and one advantage these methods would have over the mixture assignment is the use of fixed window lengths, and so a hard assignment. Of course, the advantage of the soft assignment in the Gaussian mixture models is the relatively simple probabilistic structure which we may lose if we consider alternative approaches.

The MLFM-SA model represents an attempt to construct a conditional distribution of the MLFM model which respects the structure as a transformation of the underlying random variables, and so avoids the introduction of a spurious approximation to the trajectory. Our simulation study results presented in Chapter 6 suggest that the method can perform well in the sparser sampling regions where the performance of the MLFM-AG method deteriorates. However, as we have demonstrated in 6.3.4, the MLFM-AG method is much less computationally intensive. Further work needs to be done to establish both theoretical guarantees for the consistency of this method, and the possibility of achieving greater numerical efficiency and we discuss some ideas in this direction below.

8.2 Future work

In this section we describe several directions in which we believe progress would be significant, broadly speaking these fall into either an increased theoretical understanding of the class of latent force models or increasing the range of modelling scenarios to which these methods may be applied.

8.2.1 Theoretical properties of the introduced approximations

In this thesis, we have introduced two approximate methods for fitting the latent force model with multiplicative interactions between the state and Gaussian process control functions. In Chapter 3 we used the adaptive gradient matching approximation, and in Chapter 4 we used an approximation derived from the method of successive approximations for Volterra integral equations. An important direction of future study for both of these methods would be to establish rigorous theoretical estimates of the performance in these estimators both for deriving an approximation of the distribution in state space and in carrying out a Bayesian inversion for the evolution equation. In this section, we discuss some current results and the future research directions these results suggest.

Asymptotic properties of the adaptive gradient matching approximation

We can consider the adaptive gradient matching approximation used in this work as two steps removed from the most straightforward early class of collocation methods as introduced in [Himmelblau et al., 1967, Varah, 1982]. First is the extension introduced in [Calderhead et al., 2009] which adapts these methods to the Bayesian setting, and second the approximation in [Dondelinger et al., 2013] which alters the original two-step approach, in which the trajectories are estimated with no reference to the ODE model parameters, to an adaptive approach which allows feedback between the parameter and trajectory estimates. [McGoff et al., 2015] note that even the simplest case of the two-step methods there is still relatively little known about the theoretical properties of these methods, and so it is not surprising that even less is known about the Bayesian setting and seemingly nothing about the adaptive approach. The absence of theory for the adaptive gradient matching method is related to the fact that only asymptotic consistency has been established, and once we have convergence in probability of the nonparametric estimator to the exact trajectory, the parameter based trajectory update of the adaptive gradient matching method becomes mostly redundant.

We recall from Section 1.1.1 that in general the existing theoretical guarantees are concerned with estimating the parameter vector on the basis of n observations using an objective function of the form

$$\hat{\theta}_n = \arg\min_{\theta} \int_0^1 \left\| \frac{\mathrm{d}}{\mathrm{d}\tau}\hat{\mathbf{x}}(\tau) - f(\tau, \hat{\mathbf{x}}(\tau); \boldsymbol{\theta}) \right\|^2 w(\tau)\mathrm{d}\tau, \tag{8.1}$$

for some weight function $w(\cdot)$, and where the trajectory estimate, $\hat{\mathbf{x}}$, is obtained using a plug-in kernel estimator of the function applied to the data, [Silverman, 1986]. Then under the "in-fill" asymptotic regime, that is the case in which the spacing between observations gets infinitely small as the sample size increase, this [Gugushvili and Klaassen, 2012] show that this estimator is consistent with $\sqrt{n}$-rate. While the use of estimator of the trajectory is different in this work, and the results are only established in the finite dimensional parameter setting, the general conclusion that inference is consistent as the trajectory tends towards its degenerate limit is in accordance with the results of our simulation study in Chapter 3.

Nevertheless, the efficiency of estimators of this kind are still not well understood and require a choice of the smoothing and regularisation parameters $\boldsymbol{\phi}$ and $\boldsymbol{\gamma}$ that is in general non-trivial. Given our focus on the structure preservation of the MLFM, and the relationship between the loss of accuracy of the MLFM-AG model with the average geodesic distance on the circle displayed in Figure 6.1. We have suggested that better preservation of the geometric structure, and in particular the tangent space would likely be desirable for these methods, and it may be of interest to examine the possibility of using estimates of the trajectory that preserve this geometric structure so that in (8.1) we take the integral only over trajectories on the manifold. A related idea is the concept of geometric numerical integrators [Hairer et al., 2008] for solving ODEs, and it would be useful to investigate the possibility of combining these techniques, especially if these lead to estimators with better efficiency or improved small sample asymptotics.

With regards to extending these results to the infinite dimensional setting the LFM discussed in Chapter 2 is an obvious starting point for more involved studies. The construction of the adaptive gradient matching approximation required specifying a prior Gaussian process for the trajectories. In the case of the LFM, the trajectories have a GP *conditional* on structural parameters. Therefore it is not hard to show that the adaptive gradient matching approximation to the density (3.11) is *exact* at a point $\theta_0 = \{\mathbf{D}, \mathbf{S}\}$ for the LFM.

Of course, the idea of the adaptive gradient matching process, and collocation methods more generally, is to specify a parameter independent prior. It would be interesting to establish consistency of the adaptive gradient matching methods when the GP prior on the state variables is taken from the class of LFMs with parameter different from the actual parameter and to investigate how the rate of the convergence to the true posterior is affected by this distance. With a particular focus in this investigation on the role of the spectrum of the coefficient matrix of the LFM (2.2) as this could have important connections with more general hyperbolic dynamical systems [Katok and Hasselblatt, 1995]. If it was possible to establish the results for the linear case, then a natural extension would then be to consider nonlinear models which, upon linearisation, can be described by the LFM, and to consider the use of the linearised LFM as the prior on the state when using the collocation methods to carry out inference for the nonlinear model.

Asymptotic properties of the successive approximation method

Our results in Chapter 4 began by considering the integral equation representation of the solution of a linear ODE It is therefore interesting to note that there has been some attention in the literature given to parameter estimation using this representation, rather than by matching the gradient. [Dattner and Klaassen, 2015] consider parameter inference by minimising the cost function

$$\hat{\boldsymbol{\theta}}_n = \arg\min_{\boldsymbol{\theta}} \left\| \hat{\mathbf{x}}_n(t) - \mathbf{x}_0 - \int_0^t f(\tau, \boldsymbol{\theta}, \hat{\mathbf{x}}_n) \mathrm{d}\tau \right\|^2,$$

where once again $\hat{\mathbf{x}}_n(t)$ is some plug-in estimator of the trajectory which will depend on the data.

They consider the case where the equation is linear in the parameters, that is the time-evolution equation has a representation

$$f(\mathbf{x}(t), \boldsymbol{\theta}) = \eta(\boldsymbol{\theta}) h(\mathbf{x}(t)), \tag{8.2}$$

for some functions η and h. In this case η is referred to as the *natural parameter* of the model, and they are able to prove parameter identifiability and consistency for the natural parameter as the plug-in estimator tends in probability towards its constant limit.

By considering in-fill asymptotics one is able to establish consistency as the trajectories tend towards the degenerate limit. We mentioned this same underlying idea when discussing the existence of a conditional distribution tended towards a degenerate least squares problem. However we discussed in Chapter 4 that this naturally requires completeing the trajectory in someway, and in the case when the parameter is infinite dimensional this is not straightforward.

As such our method was constructed on an assumption that this limit was unavailable, and instead we introduced the successive approximation method. Which avoided the need to introduce an ambiguous smooth parameter, although still required a regularising parameter. It would seem to us to be of immediate interest to switch to the simpler finite dimensional parameter setting and then establish in-fill consistency of the successive approximations method, in particular the realisation of the solution as a generalised least squares problem. It would also be of interest to investigate these issues from a Bayesian perspective, and to also investigate small sample asymptotics for the estimates.

Not only would this program improve our understanding of the method in general and so prepare the way for a deeper understanding of the infinite dimensional setting, but it would also allow us to investigate the extension of the successive approximation method to the case where the integral transform defining the Picard map is nonlinear. In this particular case will have the operator

$$\mathcal{K}[f](t) = \eta(\boldsymbol{\theta}) \int_{t_0}^{t} h(f(\tau))\mathrm{d}\tau,$$

where in general h is nonlinear. As such we would lose the linear dynamical system structure, and instead would have a nonlinear transformation of each approximation, however it may still be feasible to consider the extended and unscented Kalman filter, [Kalman and Bucy, 1961, Julier and Uhlmann, 2004], approximations to this nonlinear system.

8.2.2 Local experts for the method of successive approximations

In Section 4.5 we introduced the MLFM-MixSA model, which combined local versions of the MLFM-SA model to counteract the decay in the accuracy of the approximation over longer intervals. The result was a Gaussian process mixture model, and our simulation study in Section 6.2.6 suggests that this approach has the potential to lead to more accurate inference.

However, it is clear that more work needs to be done in understanding this method. Previously, we remarked on the possibility of using convolution layers in the neural network setup of Section 4.4.1 to achieve the desired local approximations. An alternative, but likely related, construction is given by *local polynomial regression*, [Stone, 1977, Cleveland, 1979]. These models are designed to learn the non-parameteric mean function $\mu(\cdot)$ in the regression model

$$Y(t_i) = \mu(t_i) + \boldsymbol{\epsilon}_i, \tag{8.3}$$

for independent i.i.d errors $\boldsymbol{\epsilon}_i$. The idea is to expand the mean function by its Mth order Taylor series

$$\mu(t_i) = \sum_{m=0}^{M} \frac{(t-t_0)^m}{m!}\mu^{(m)}(t_0),$$

where $\mu^{(m)}(t_0)$ is the mth order derivative of μ evaluated at t_0. Since the Taylor series provides only a local approximation the idea is to approximate this expansion by finding the vector $\hat{\boldsymbol{\beta}} \in \mathbb{R}^{M+1}$ which minimises the objective function

$$\sum_{i=1}^{N} \left(Y_i - \sum_{m=0}^{M} \beta_m (t-t_0)^m \right)^2 K(t_i - t_0), \tag{8.4}$$

for some appropriate choice of kernel function $K(\cdot)$. Under certain assumptions it can be shown that

$$\hat{\beta}_m \xrightarrow{\mathbb{P}} \frac{\mu^{(m)}(t_0)}{m!},$$

where the convergence is taken to be convergence in probability.

In the case of linear ODEs with finite dimensional parameters then one may easily show that the approximation obtained by repeated iteration of the Picard map agrees with the Taylor series expansion of the solution. This naturally suggests the use of a local polynomial estimator to carry out inference. Indeed we could derive a local polynomial estimator for each component of the trajectory, $x_k(t)$, $k = 1, \ldots, K$, ariving at a collection of coefficients $\hat{\beta}_{km}$, for $k = 1, \ldots, K$ and $m = 0, 1, \ldots, M$. We could then attempt to recover the true parameters by matching these coefficients with the polynomial expansion given by the truncated Neumann series. Whether the parameter is identifiable will then depend on a system of polynomials which can be checked, although in practice this may be complicated to do, nevertheless this is equally true for many other conditions for parameter identification in dynamic systems models.

After carrying out the full derivation for the finite dimensional linear case, and using existence results for local polynomial regression to establish consistency and distributional properties, then a natural extension would be to consider the possiblity of applying this method to systems which are linear in parameters as discussed in the previous section, and the infinite dimensional setting. Knowledge of small sample asymptotics and optimal selection of the bandwidth in the linear case would likely lead to a much improved understanding of the linear infinite dimensional case, and how we might go about constructing a mixture of the MLFM-SA model.

8.2.3 Latent manifold exploration

The ultimate purpose of our extension of the LFM framework in Chapter 2 to now include multiplicative interactions between the latent force and state variables was to enable the embedding of known geometric constraints into the model building process while continuing to avoid the need to provide a complete mechanistic specification of the dynamic system. This point of view guided our choice of application in Chapter 7 to human motion data for which the fixed length of joint segments, and the recording of motion as relative rotations, naturally suggested an appropriate decomposition of the data space.

As we consider future applications and seek to expand the list of modelling scenarios in which our proposed method could be utilised, we would like to relax this assumption of a known geometric constraint. Instead, we would want to allow for the topological structure of the latent variable space to be learned automatically, and for the possibility of this step being combined with a more substantial dimensionality reduction. As such there are two challenges which must be addressed, the first is the process of manifold exploration in either the data space or latent variable space, and the second is combining this exploration with a dimensionality reduction technique.

If we first consider the process of manifold exploration in a given space, then one option open to us would be to fit the model using the methods developed in this thesis for a wide range of different manifold structure and then use some form of model validation process to select the model. Rather than perform model selection after fitting the model a more satisfying method would allow for the construction on an inferential process with the ability to move between different manifolds during the learning process, and so marginalise over the various possible choices of geometric structure.

In general moving between manifolds during the learning process is a hard problem because transformations between manifolds are typically nonlinear. Furthermore if we want to allow for interesting moves we must need consider non-invertible maps; otherwise we would be exploring manifolds that are equivalent up to a continuous bijection, that is to say, manifolds which are homeomorphic.

So for the model to be attractive, it must move between manifolds which are not homeomorphic and therefore we must consider transformations that are not continuous bijections. As a result, we cannot make immediate use of the Jacobian as a means of weighting the change of the probability with the transformation.

However, while the transformations between the manifolds are necessarily complex an attractive feature of the MLFM framework which we have introduced is that the manifold constraints are encoded in the model by the Lie algebra determining the support of the matrix-valued process $\mathbf{A}(t)$ so that differing Lie algebras represent different geometries. In contrast to the transformation between different manifolds the move between Lie algebras is a move from one finite-dimensional vector space to a new (finite dimensional) vector space, and therefore it is simpler to understand how the model transforms.

If we would like to consider different dimensions of the latent variable space, then the transformations between the different Lie algebra structures will typically not be invertible, but the vector space setting makes it simpler to place the different possible model configurations inside a single space and so better understand how we might move between models.

To make this explicit we first fix a dimension of the space n, then for a given model the matrix $\mathbf{A}(t) \in \mathbb{R}^{n\times n}$ will be a member of the Lie-algebra $\mathfrak{gl}_n$, which is the n^2-dimensional Lie algebra generating the group of invertible linear transformations on $\mathbb{R}^n$, usually referred to as the general linear group. This Lie algebra has a basis

$$\mathbf{E}_{ij}^{(n)} = \{\mathbf{M} \in \mathbb{R}^{n\times n} \ : \ [\mathbf{E}_{ij}^{(n)}]_{kl} = \delta_{ik}\delta_{jl}\}.$$

Therefore every possible specification of the MLFM for a fixed dimension is expressable with respect to this basis, and so using the notation of Section 2.3 we would have $D = n^2$, and a set of basis matrices $\{\mathbf{L}_d\}_{d=1}^{D}$ by choosing an enumeration of the set of matrices $\mathbf{E}_{ij}$ for $i, j = 1, \ldots, n$. So that the methods introduced in this thesis already allow for the integration over all such possible specifications if we pick a suitably large set of basis matrices. The quadratic scaling of the size of basis matrices, and so the number of connection parameters β_{rd} with the dimension of the state variable is, of course, a possible cause for concern, however our objective at this stage is to work towards a significant dimension reduction, so this is not necessarily prohibitive.

A somewhat more subtle issue is whether it is best to optimise over the complete set of basis matrices freely, or whether a more controlled search of the model space might be preferred. This issue deserves some attention, for instance, we could imagine a Monte-Carlo search of the model space proceeding by adding and removing basis matrices, but also with more structured changes such as proposing the addition of asymmetric, or skew-symmetric pairs of basis matrices.

If we now allow the dimension of the state space to vary, typically we would want to achieve a dimension reduction and so if K continues to denote the dimension of the dataspace then we can consider the case of all possible specifications of the coefficient matrix on the space

$$\mathcal{M} = \{\mathfrak{gl}_n\}_{n=1}^{K}. \tag{8.5}$$

In terms of moving from $\mathfrak{gl}_n$ to $\mathfrak{gl}_{n+1}$ a natural choice would be to consider the mapping of each basis matrix $\mathbf{E}_{ij}^{(n)} \mapsto \mathbf{E}_{ij}^{(n+1)}$, which corresponds to nesting each of the $n \times n$ matrices as a block in the larger $(n+1)\times(n+1)$, with a similar idea for moving down in dimension. A full description of this process is the subject of future research, but a graphical representation of the idea is given in Figure 8.1. Since the problem reduces to moving between finite dimensional vector spaces a promising choice of learning algorithm is given by reversible MCMC techniques [Green, 1995].

We must now consider how to carry out the transformation from the latent variable space and the data space. In the case that this relationship is linear, then the inference is likely to remain relatively straightforward. Because we would like to retain a probabilistic structure to our approximations, then a natural place to start is the use of the GP latent variable model [Lawrence, 2003].

Existing approaches to dimensionality reduction that have used the GP-LVM, or similar frameworks, includes the BC-GP-LVM which we mentioned in the previous section, the GPDM described in Section 1.1.6 and the topologically constrained latent variable model TC-LVM [Urtasun et al., 2008] and the GPDM [Wang et al., 2006].

In general, these methods have the property that the data conditional on the latent states, is a GP, but that the kernel of this conditional GP depends in a nonlinear way on the state variables. This is very similar to the situation encountered for the conditional distribution of the state variable in the MLFM-AG and MLFM-SA approximations; however, the difference in these situations is that each entry of the covariance function typically depends on the *full* set of latent states. In comparison, the entries of the covariance matrix in the GP-LVM depend only on the latent states at two distinct points.

While this makes model somewhat more complex, it does not seem completely prohibative. Furthermore the variational methods developed for the GP-LVM in [Titsias and Lawrence, 2010] only require the ability to take the expectation of certain statistics of the covariance function, and so while they would be more complex in the fully dependent specifcation of the covariance function they do not seem beyond the realms of possibility.

More generally for the methods discussed above it is not immediately obvious what any of the global geometric or topological structure of the resulting latent space might be. In contrast for the MLFM framework which we have introduced the topology is well understood being composed of products of well studied matrix Lie groups. Therefore if we can combine the Gaussian conditional structure of our approximations to the MLFM with a signficiant dimensionality reductions then we will be able to carry out learning of highly structured low dimensional latent spaces, which in contrast to the TC-LVM do not attempt to embed topological constraints through the covariance function, but rather the much more flexible choice of the coefficient matrix.

8.2.4 Parameter-driven models

For dynamical systems models with time-varying parameters [Cox et al., 1981] makes a distinction between those which may be regarded as *observation driven*, and those which are *parameter driven*. In both cases one assumes that the output, $\mathbf{y}_s$, of the process at a discrete, ordered, set of index times $s \in \{0, 1, \ldots, T\}$ is the output of an, ideally exponential family, distribution governed by the time-dependent parameter ϕ_s.

In the case of the observation-driven model, the current parameter estimates are assumed to be given by deterministic functions of the output dependent variables, as well any additional covariates available up to, and including, time s. Let $\mathbf{Y}^{(s-1)} = (\mathbf{y}_{s-1}, \mathbf{y}_{s-2}, \ldots)$ denote the history of the process up to index s then we write $p(y_s \mid$

$\mathbf{Y}^{(s-1)}) := p(\mathbf{y}_s \mid \phi_s)$ with a parameter of the form

$$\phi_s = \phi(\mathbf{Y}^{(s-1)}, w_s), \tag{8.6}$$

where w_s is the output of some random Markovian innovation process at time s, and the function $\phi(\cdot)$ is a known deterministic function. Under this paradigm the parameters evolve randomly over time, but are perfectly predictable one-step-ahead given the past information.

Alternatively, within the parameter-driven, paradigm one instead models the parameter as

$$\phi_s = \phi^{\dagger}(\phi^{(s-1)}, w_s^{\dagger}), \tag{8.7}$$

for some function $\phi^{\dagger}(\cdot)$, and where again $w_s^{\dagger}$ is a pure noise innovation process. Important special instances of such models include the stochastic volatility models [Tauchen and Pitts, 1983] which have seen widespread use in finance and economics. Under this setting the parameters themselves are allowed to vary over time as a stochastic processes, and for this reason [Cox et al., 1981] referred to them as *latent structure models* and we now consider the connection between these models and the latent force models considered in this thesis.

Linear latent force models

We first consider the linear latent force model discussed in Section 2.2, and for simplicity we shall illustrate our discussion with the simplest example of a univariate LLFM with a single driving force

$$\frac{\mathrm{d}x(t)}{\mathrm{d}t} = -Dx(t) + Sg(t). \tag{8.8}$$

We further assume that the driving force, $g(t)$, has a representation as a stochastic differential equation, this is true for many popular GPs with stationary kernel functions [Säarkä and Solin, 2019], then we may write a state space representation of this GP as

$$g(t) = \mathbf{H}\mathbf{z} \tag{8.9a}$$

$$\mathrm{d}\mathbf{z} = \mathbf{F}\mathbf{z}\mathrm{d}t + \mathbf{L}\mathrm{d}\mathbf{w}(t), \tag{8.9b}$$

for some observation matrix $\mathbf{H}$, drift matrix $\mathbf{F}$ and diffusion matrix $\mathbf{L}$, where $\mathbf{w}(t)$ is a vector-valued, but possibly degenerate, white noise process.

It follows that the augmented process $\tilde{\mathbf{z}}(t) = (x(t), \mathbf{z}(t))^{\top}$ will also have a linear state space representation, and in particular after discretisation we can write the process as a linear Gaussian dynamical system, with a Gaussian emission density and a latent Markov structure. From this we can easily relate parameter driven models and the LFM by identifying the time varying parameter with the state space process $\tilde{\mathbf{z}}$ at discrete times, and then choosing an emission density, in particular if this emission density is Guassian we can fit these models using standard Kalman filtering methods.

Multiplicative latent force models

The discussion in [Cox et al., 1981] makes it clear than the distinction between the parameter and observation driven setup becomes more interesting in the case of non-Gaussian time series, and we now consider how we would have to adapt the above setup if we wish to add multiplicative interactions.

We shall again assume that the driving GP has a state space representation (8.9), and if we consider a simple example of the MLFM with a single latent force, then we obtain a system of stochastic differential equations

$$\mathrm{d}\mathbf{x}(t) = \left[\mathbf{A}_0 + (\mathbf{H}\mathbf{z}(t)) \cdot \mathbf{A}_1\right] \mathbf{x}(t)\mathrm{d}t \tag{8.10a}$$

$$\mathrm{d}\mathbf{z}(t) = \mathbf{F}\mathbf{z}\mathrm{d}t + \mathbf{L}\mathrm{d}\mathbf{w}(t). \tag{8.10b}$$

After discretisation this becomes a nonlinear state space model, with non-Gaussian trajectories which is one of the most general methods of constructing parameter driven models, [Koopman et al., 2016]. As such the MLFM may be regarded as a particular instance of a parameter driven model. However, as made clear from our discussion of the Lie algebra/group structure in Section 2.3 there is important invariant structure that should be taken into account when performing the discretisation to construct the corresponding discrete time parameter-driven model. In particular we recall that after discretisation the MLFM can be written as a sequence of sequential transformations

$$\mathbf{x}_s = \mathcal{R}(\{g(t) \ : t \in [t_{s-1}, t_s]\})\mathbf{x}_{s-1}, \tag{8.11}$$

where $\mathcal{R}$ is an element of some Lie group G. This element depends only on the latent force over the interval $[t_{s-1}, t_s]$, and formally we can write

$$R(\{g(t) \ : t \in [t_{s-1}, t_s]\}) = \mathcal{T} \exp\left(\int_{t_{s-1}}^{t_s} g(\tau)\mathrm{d}\tau \right), \tag{8.12}$$

where $\mathcal{T}$ is the "time-ordering" operation [van Kampen, 2007] which rewrites (8.12) as the formal solution (4.9) to the MLFM given in Section 4.2. This is now a non-linear, non-Gaussian state space model and it not necessarily clear how one might perform inference in this model. The representation (8.12) is only a formal one, and some approximation be done such the one we have developed in Chapter 4. One possibility would be to consider using extended Kalman filtering ideas for the nonlinear continuous time state space model (8.10) such as in [Hartikainen et al., 2012], but a naive application of these ideas would ignore the special nature of the Lie group structure.

An interesting alternative would be to attempt to construct a Markov process in the Lie algebra itself, rather than the specification in (8.10) which constructs the state space model on the group itself – a similar idea is contained in the specification of the function $\Omega(t)$ in the Magnus series solution discussed in Section 4.2.1 and further exploration of this deserves attention. Once a suitable time-varying system has been constructed in the Lie algebra one can then consider emissions which take the form of exponential family distributions which are supported on the relevant Lie group, in simple examples such as the circle such emission distributions would coincide with the wrapped distributions used in circular statistics and their higher dimensional analogues [Mardia and Jupp, 2000]. On a potentially related note [Bather, 1965] provided some conditions for the existence of convenient summarising statistic for the parameter chain $\{\phi_s\}_{s=0}^{T}$, so that inference may be done efficiently, without growing in dimension as the number of time-steps increases. These conditions however require solving a certain integral equation which in general is likely to be prohibitively challenging. However, it may be possible that invariant solutions to this integral equation can be obtained for the special case of the MLFM by using symmetry properties of the Lie groups themselves, and we leave this investigation to future work.

8.2.5 Partial differential equations

An important extension of the models introduced in this thesis, which have been constructed around ODEs, would be to systems described by partial differential equations (PDEs) which model an output $\mathbf{y}(t, \mathbf{x})$, depending on both a temporal argument and an M-dimensional spatial argument, $\mathbf{x}$, using an evolution equation of the form

$$\frac{\partial \mathbf{y}(t,\mathbf{x})}{\partial t} = f\left(t, \mathbf{x}, \mathbf{y}, \frac{\partial \mathbf{y}}{\partial x_1}, \ldots, \frac{\partial \mathbf{y}}{\partial x_N}, \frac{\partial^2 \mathbf{y}}{\partial x_1 \partial x_1}, \ldots, \frac{\partial^2 \mathbf{y}}{\partial x_1 \partial x_M}, \ldots\right). \tag{8.13}$$

To remain within the linear latent force model framework we would have to assume that this evolution equation is linear with respect to the output variable and its mixed derivatives. The application of the LFM described in Chapter 2 to such a system has been considered in [Alvarez et al., 2009] for the heat equation

$$\frac{\partial y_k(t)}{\partial t} = \sum_{j=1}^{M} \kappa_k \frac{\partial^2 y_k(t,\mathbf{x})}{\partial x_j^2} + \sum_{r=1}^{R} S_{rk} g_r(t), \qquad k = 1, \ldots, K.$$

Note once again the diagonal structure of this model ensuring that interactions between the field variables, $y_q(t, \mathbf{x})$, occur only through the set of common latent forces, this is required to ensure tractability of integrals defining the covariance of the solution similar to the case described in Section 2.2.1. For a model of this form, it is possible to express the solution as a convolution of the latent force with a Greens' function associated with the PDE, again similar to the convolution operators (2.7) in Section 2.2.1, leading to an explicit expression for the covariance function and a preservation of the GP regression framework.

Since, at least in relatively simple cases, it is possible to extend the LFM to the PDE setting it is reasonable to consider whether this remains true if we now allow for multiplicative interactions between the latent forces and the output variables along the lines of the MLFM introduced in this thesis. Once we allow for multiplicative interactions, it will typically no longer be the case that we can obtain an explicit Greens' function for the solution, mirroring our discussion on the absence of a simple expression for the fundamental solution of the MLFM. One possible way of proceeding would be to discretise the spatial variable, and so recast the PDE as a system of ODEs; this is referred to as the 'method of lines' [Schiesser, 1991]. Once we have carried out this process, we could, in principle, apply the techniques developed in this thesis to this extensive system of ODEs, which will be of the MLFM form. However in practice, there are several difficulties in carrying over the procedures already developed in such a direct way, the most serious of these difficulties are associated with the fact that the dimension of the state variable, K, will now be of a very high dimension.

For example in the MLFM-AG method which requires introducing an independent GP interpolant to each dimension of the state variable, we would now have to perform K inversions of the $N \times N$ covariance matrices which will quickly become computationally demanding. One option to rescue the method in this setting would be to insist that all of the GPs have the same kernel function and hyperparameter. However, this seems very unsatisfying given the dependence of the GP interpolants on these hyperparameters which we have discussed in this thesis and the fact that variations of the functions described by PDEs at different spatial coordinates are likely to be qualitatively very different.

The situation is no better for the case of the MLFM-SA method where the transformation operator is of size $(NK)^2$ so that with very high dimensional state variables this method will not be possible. The situation is not entirely hopeless because the conversion of the righthand side of (8.13) to a discretised form using finite differences leads to a transformation matrix with a high degree of sparsity. This sparsity will further carry on into the construction of the discrete version of the Picard operator for the semi-discretised PDE, however, there is still a significant challenge in implementing such a method correctly.

For both of these approaches, the sparsity of the discretised form of the equation also implies specific conditional independence properties of the semi-discretised variables $\mathbf{y}(t, x_i)$. The process of choosing an inference strategy which respects this conditional independence structure referred to as 'coding' in [Besag, 1974] for spatial random fields. Successful coding is likely to be a necessary step in making these methods practical, however, for both methods, there is the further issue of potential problems regarding the stability of the chosen discretisation scheme.

Proper orthogonal decompositions

In the context of PDEs and the associated issues of high dimensionality, it is worth considering a popular dimension reduction strategy is popular in physics referred to as 'proper orthogonal decompositions', but more familiar in the statistics and machine learning community as principal component analysis. In this method, a set of N observations of a K dimensional variable are represented by linear combinations of a set of ℓ dimensional basis vectors, with $\ell < \min\{N, K\}$ in an optimal manner.

$$\min_{\psi_1,\ldots,\psi_\ell \in \mathbb{R}^K} \int_0^T \left\| \mathbf{x}(t) - \sum_{i=1}^{\ell} \langle \mathbf{x}(t), \psi_i \rangle_W \psi_i \right\|_W^2 \mathrm{d}t, \quad \text{s.t.} \quad \langle \psi_i, \psi_j \rangle_W = \delta_{ij}, 1 \leq i, j \leq \ell,$$

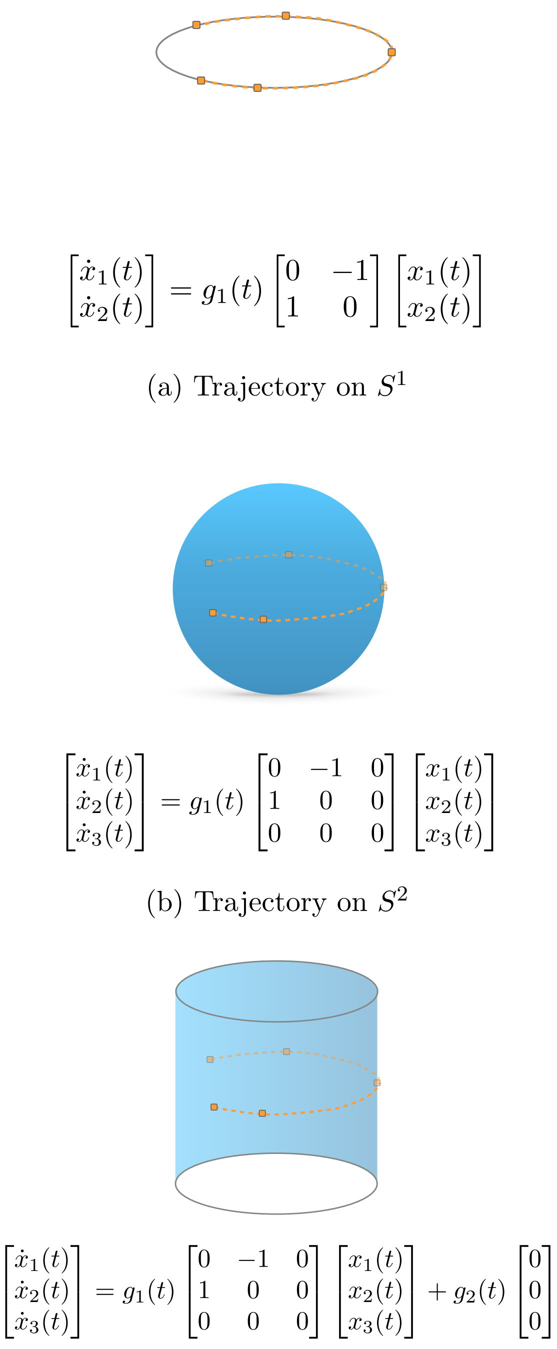
here we use the bracket notation to denote a, possibly weighted, inner product on $\mathbb{R}^K$ given by $\langle \mathbf{x}, \mathbf{y} \rangle_W = \mathbf{x}^\top W \mathbf{y}$ for vectors $\mathbf{x}, \mathbf{y} \in \mathbb{R}^K$ and symmetric positive definite matrix W. This objective function is typically replaced by a dense sample of the process $\mathbf{x}(t)$, referred to in the POD literature as 'snapshots', and replacing the integral with a numerical quadrature. As in standard PCA, it is not hard to show that the optimal choice of basis vectors are given by weighted versions of the eigenvectors of the sample covariance matrix. Indeed if we combine the quadrate weights $\{\alpha_j\}$ into the $N \times N$ diagonal matrix D, the collection of observations of the states into $\mathbf{X}$ then if $\mathbf{U}, \mathbf{\Sigma}, \mathbf{V}$ is the singular value decomposition of $\bar{\mathbf{X}}$ then the basis vectors are given as in standard PCA by

$$\psi_i = W^{1/2} \mathbf{U}_{:,i}.$$

Little attention in the literature has been given to the probabilistic interpretation of the POD method. However, it has been noted [Tipping and Bishop, 1999, Bishop, 1999] that PCA can be given a probabilistic interpretation as a Gaussian latent variable model. With this in mind, it would be of immediate interest to compare the Gaussian approximation arising from the POD model reduction of the LFM discussed in Chapter 2 with the exact Gaussian distribution, and to further compare this with the Gaussian approximation provided by the adaptive gradient matching methods.

As we saw in Chapter 3 the MLFM-AG approximation allowed for a convenient conditional Gaussian structure providing tractable parameter inference, whereas for the POD method the dependence of the parameters will get lost in the construction of the singular value/eigenvalue decomposition of the model matrices.

Since the POD method leads to very efficient dimensionality reduction, and the MLFM-AG methods lead to very tractable conditional inference it would be an interesting research program to investigate the linear case in order to achieve a more in-depth understanding on the connection between these two approaches. Moreover, we could then investigate the possibility of creating a hybrid method attempting to embody the best aspects of both methods, leading to significant dimensionality reduction and tractable parameter inference in very high dimensional dynamical systems as an alternative to the GP based dimensionality reduction techniques we discussed above.



$$\begin{bmatrix}\dot{x}_1(t)\\ \dot{x}_2(t)\end{bmatrix} = g_1(t)\begin{bmatrix}0 & -1\\ 1 & 0\end{bmatrix}\begin{bmatrix}x_1(t)\\ x_2(t)\end{bmatrix}$$

(a) Trajectory on S^1

$$\begin{bmatrix}\dot{x}_1(t)\\ \dot{x}_2(t)\\ \dot{x}_3(t)\end{bmatrix} = g_1(t)\begin{bmatrix}0 & -1 & 0\\ 1 & 0 & 0\\ 0 & 0 & 0\end{bmatrix}\begin{bmatrix}x_1(t)\\ x_2(t)\\ x_3(t)\end{bmatrix}$$

(b) Trajectory on S^2

$$\begin{bmatrix}\dot{x}_1(t)\\ \dot{x}_2(t)\\ \dot{x}_3(t)\end{bmatrix} = g_1(t)\begin{bmatrix}0 & -1 & 0\\ 1 & 0 & 0\\ 0 & 0 & 0\end{bmatrix}\begin{bmatrix}x_1(t)\\ x_2(t)\\ x_3(t)\end{bmatrix} + g_2(t)\begin{bmatrix}0\\ 0\\ 0\end{bmatrix}$$

(c) Trajectory on $S^1 \times \mathbb{R}$

Figure 8.1: Graphical representation of embedding a single trajectory of the MLFM into different manifolds by varying the coefficient matrix. In each figure the model has the same trajectory, but we can change the space of possible trajectories realised. The initial example on the circle (a) can be considered as a model on S^2 by nesting the coefficient matrix in a higher dimensional vector space (b), or nested in the cylinder by also allowing for affine translations in (c).

Bibliography

Martín Abadi, Ashish Agarwal, Paul Barham, Eugene Brevdo, Zhifeng Chen, Craig Citro, Greg S. Corrado, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Ian Goodfellow, Andrew Harp, Geoffrey Irving, Michael Isard, Yangqing Jia, Rafal Jozefowicz, Lukasz Kaiser, Manjunath Kudlur, Josh Levenberg, Dandelion Mané, Rajat Monga, Sherry Moore, Derek Murray, Chris Olah, Mike Schuster, Jonathon Shlens, Benoit Steiner, Ilya Sutskever, Kunal Talwar, Paul Tucker, Vincent Vanhoucke, Vijay Vasudevan, Fernanda Viégas, Oriol Vinyals, Pete Warden, Martin Wattenberg, Martin Wicke, Yuan Yu, and Xiaoqiang Zheng. TensorFlow: Large-scale machine learning on heterogeneous systems, 2015. Software available from tensorflow.org.

Terrence M Adams and Andrew B Nobel. Finitary reconstruction of a measure preserving transformation. *Israel J. Math.*, 126(1):309–326, December 2001.

R. J. Adler and Jonathan E. Taylor. *Random Fields and Geometry*. Springer Monographs in Mathematics. Springer, New York, 2007.

Mauricio Alvarez, David Luengo, and Neil Lawrence. Latent force models. In David van Dyk and Max Welling, editors, *Proceedings of the Twelth International Conference on Artificial Intelligence and Statistics*, volume 5 of *Proceedings of Machine Learning Research*, pages 9–16, Hilton Clearwater Beach Resort, Clearwater Beach, Florida USA, 16–18 Apr 2009. PMLR.

K. N. An. Kinematic analysis of human movement. *Annals of Biomedical Engineering*, 12:585–597, 1984.

Nairo D. Aparicio, Simon J. A. Malham, and Marcel Oliver. Numerical evaluation of the Evans function by Magnus integration. *BIT Numerical Mathematics*, 45(2): 219–258, Jun 2005. ISSN 1572-9125. doi: 10.1007/s10543-005-0001-8.

F. T. Arechhi and R. Bonifacio. Theory of optical maser amplifiers. *IEEE Journal of Quantum Electronics*, 1:169–178, 1965.

Vladimir Arnold. *Ordinary Differential Equations*. MIT Press, Cambridge, Massachusetts, 1973.

Kendall E. Atkinson. *The Numerical Solution of Integral Equations of the Second Kind*. Cambridge Monographs on Applied and Computational Mathematics. Cambridge University Press, Cambridge, 1997. doi: 10.1017/CBO9780511626340.

Christopher T.H. Baker. A perspective on the numerical treatment of Volterra equations. *Journal of Computational and Applied Mathematics*, 125(1):217 – 249, 2000. ISSN 0377-0427. doi: https://doi.org/10.1016/S0377-0427(00)00470-2. Numerical Analysis 2000. Vol. VI: Ordinary Differential Equations and Integral Equations.

David Barber and Yali Wang. Gaussian processes for bayesian estimation in ordinary differential equations. In Eric P. Xing and Tony Jebara, editors, *Proceedings of the 31st International Conference on Machine Learning*, volume 32 of *Proceedings of Machine Learning Research*, pages 1485–1493, Bejing, China, 22–24 Jun 2014. PMLR. URL http://proceedings.mlr.press/v32/barber14.html.

J. A. Bather. Invariant conditional distributions. *Ann. Math. Statist.*, 36(3):829–846, 06 1965. doi: 10.1214/aoms/1177700057. URL https://doi.org/10.1214/aoms/1177700057.

M. A. Beaumont, W. Zhang, and D. J. Balding. Approximate Bayesian computation in population genetics. *Genetics*, 162(4):2025–2035, 2002.

R. Bellman and K.J. Åström. On structural identifiability. *Mathematical Biosciences*, 7(3):329–339, 1970. ISSN 0025-5564. doi: https://doi.org/10.1016/0025-5564(70)90132-X.

L. Mark Berliner. Physical-statistical modeling in geophysics. *Journal of Geophysical Research: Atmospheres*, 108(24), 2003. doi: 10.1029/2002JD002865.

J. Besag. Comments on "representations of knowledge in complex systems" by U. Grenander and M. I. Miller. *Journal of the Royal Statistical Society, Series B*, 56: 591–592, 1994.

Julian Besag. Spatial interaction and the statistical analysis of lattice systems. *Journal of the Royal Statistical Society. Series B (Statistical Methodology)*, 36(2):192–236, 1974. ISSN 00359246.

Alexandros Beskos, Omiros Papaspiliopoulos, Gareth O Roberts, and Paul Fearnhead. Exact and computationally efficient Likelihood-Based estimation for discretely observed diffusion processes. *Journal of the Royal Statistical Society. Series B (Statistical Methodology)*, 68(3):333–382, 2006.

Christopher M. Bishop. *Neural Networks for Pattern Recognition.* Oxford University Press, Oxford, 1995.

Christopher M. Bishop. Bayesian PCA. In M. J. Kearns, S. A. Solla, and D. A. Cohn, editors, *Advances in Neural Information Processing Systems 11*, pages 382–388. MIT Press, 1999.

Christopher M. Bishop. *Pattern Recognition and Machine Learning.* Springer-Verlag, Berlin, Heidelberg, 2006. ISBN 0387310738.

J. P. N. Bishwal. *Parameter Estimation in Stochastic Differential Equations.* Springer, New York, 2008.

A. Bissacco. Modeling and learning contact dynamics in human motion. In *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, volume 1, pages 421–428, 2005. doi: 10.1109/CVPR.2005.225.

F. C. Boogerd, F. J. Bruggeman, and R. C. Richardson. Mechanistic explanations and models in molecular systems biology. *Foundations of Science*, 18:725–744, 2013.

Stephen P. Brooks and Andrew Gelman. General methods for monitoring convergence of iterative simulations. *Journal of Computational and Graphical Statistics*, 7(4):434–455, 1998. doi: 10.1080/10618600.1998.10474787. URL `https://www.tandfonline.com/doi/abs/10.1080/10618600.1998.10474787`.

W. L. Buntine. Operations for learning with graphical models. *Journal of Artificial Intelligence Research*, 2:159–225, 1994.

Ben Calderhead, Mark Girolami, and Neil D. Lawrence. Accelerating bayesian inference over nonlinear differential equations with gaussian processes. In D. Koller, D. Schuurmans, Y. Bengio, and L. Bottou, editors, *Advances in Neural Information Processing Systems 21*, pages 217–224. Curran Associates, Inc., 2009.

W. Cleveland. Robust locally weighted regression and smoothing scatterplots. *Journal of the American Statistical Association*, 74(368):829–836, 1979.

D. R. Cox and H. S. Battey. Large numbers of explanatory variables, a semi-descriptive analysis. *Proceedings of the National Academy of Sciences*, 114(32):8592–8595, 2017. ISSN 0027-8424. doi: 10.1073/pnas.1703764114.

D. R. Cox, Gudmundur Gudmundsson, Georg Lindgren, Lennart Bondesson, Erik Harsaae, Petter Laake, Katarina Juselius, and Steffen L. Lauritzen. Statistical analysis of time series: Some recent developments [with discussion and reply]. *Scandinavian Journal of Statistics*, 8(2):93–115, 1981. ISSN 03036898, 14679469. URL `http://www.jstor.org/stable/4615819`.

Walter J. Culver. On the existence and uniqueness of the real logarithm of a matrix. *Proceedings of the American Mathematical Society*, 17(5):1146–1151, 1966.

Itai Dattner and Chris A J Klaassen. Optimal rate of direct estimators in systems of ordinary differential equations linear in functions of the parameters. *Electronic Journal of Statistics*, 9(2):1939–1973, 2015.

A. P. Dempster, N. M. Laird, and D. B. Rudin. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society, Series B*, 39(1):1–38, 1977.

S. N. Dixit, P. Zoller, and P. Lambropoulos. Non-Lorentzian laser line shapes and the reversed peak asymmetry in double optical resonance. *Physical Review A*, 21:1289–1296, Apr 1980. doi: 10.1103/PhysRevA.21.1289.

Frank Dondelinger, Dirk Husmeier, Simon Rogers, and Maurizio Filippone. Ode parameter inference using adaptive gradient matching with Gaussian processes. In Carlos M. Carvalho and Pradeep Ravikumar, editors, *Proceedings of the Sixteenth International Conference on Artificial Intelligence and Statistics*, volume 31 of *Proceedings of Machine Learning Research*, pages 216–228, Scottsdale, Arizona, USA, 29 Apr–01 May 2013. PMLR.

Richard C. Dorf and Robert H. Bishop. *Modern Control Systems*. Prentice Hall, New York, twelth edition, 2011.

D.C Dowson and B.V Landau. The Fréchet distance between multivariate normal distributions. *Journal of Multivariate Analysis*, 12(3):450–455, 1982. ISSN 0047-259X. doi: https://doi.org/10.1016/0047-259X(82)90077-X.

Yong Du, W. Wang, and L. Wang. Hierarchical recurrent neural network for skeleton based action recognition. In *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1110–1118, June 2015. doi: 10.1109/CVPR.2015.7298714.

Simone Duane, A. D. Kennedy, Brian J. Pendleton, and Duncan Roweth. Hybrid Monte Carlo. *Physics Letters B*, 195(2):216–222, 1987.

F. J. Dyson. The radiation theories of Tomonaga, Schwinger, and Feynman. *Physical Review*, 75:486–502, 1949. doi: 10.1103/PhysRev.75.486.

B. S. Everitt. *An Introduction to Latent Variable Models.* Monographs on Statistics and Applied Probability. Chapman and Hall, London, 1984.

Ronald Fisher. On the mathematical foundations of theoretical statistics. *Philosophical Transactions of the Royal Society of London A: Mathematical, Physical and Engineering Sciences*, 222(594-604):309–368, 1922. ISSN 0264-3952. doi: 10.1098/rsta.1922.0009.

Wendell H. Fleming and Raymond W. Rishel. *Deterministic and Stochastic Optimal Control.* Springer-Verlag, New York, 1975.

K. Fragkiadaki, S. Levine, P. Felsen, and J. Malik. Recurrent network models for human dynamics. In *2015 IEEE International Conference on Computer Vision (ICCV)*, pages 4346–4354, Dec 2015. doi: 10.1109/ICCV.2015.494.

F. R. Gantmacher. *The Theory of Matrices.* Chelsea Publishing Company, New York, 1959.

Andrew Gelman and Donald B. Rubin. Inference from iterative simulation using multiple sequences. *Statist. Sci.*, 7(4):457–472, 11 1992. doi: 10.1214/ss/1177011136. URL `https://doi.org/10.1214/ss/1177011136`.

Peter J. Green. Reversible jump markov chain monte carlo computation and bayesian model determination. *Biometrika*, 82(4):711–732, 1995. doi: 10.1093/biomet/82.4.711.

Shota Gugushvili and Chris A J Klaassen. $\sqrt{}$n-consistent parameter estimation for systems of ordinary differential equations: bypassing numerical integration via smoothing. *Bernoulli*, 18(3):1061–1098, August 2012.

Ernst Hairer, Christian Lubich, and Gerhard Wanner. *Geometric Numerical Integration.* Springer Series in Computational Mathematics. Springer, Berlin, 2008.

Brian C. Hall. *Lie Groups, Lie Algebras and Representations.* Graduate Texts in Mathematics. Springer, New York, 2015.

Jouni Hartikainen, Mari Seppänen, and Simo Särkkä. State-space inference for nonlinear latent force models with application to satellite orbit prediction. In *Proceedings of the 29th International Conference on Machine Learning*, volume abs/1206.4670 of *ICML '12*, 2012.

Trevor Hastie, Robert Tibshirani, and Jerome Friedman. *The Elements of Statistical Learning.* Statistics. Springer, New York, 2009.

W. K. Hastings. Monte Carlo sampling methods using Markov chains and their applications. *Biometrika*, 57(1):97–109, 1970. ISSN 00063444.

S. Helgason. *Differential Geometry and Symmetric Spaces.* Academic Press, New York, 1962.

D. M. Himmelblau, C. R. Jones, and K. B. Bischoff. Determination of rate constants for complex kinetics models. *Industrial & Engineering Chemistry Fundamentals*, 6 (4):539–543, 1967. doi: 10.1021/i160024a008.

Geoffrey E. Hinton. Training products of experts by minimizing contrastive divergence. *Neural Computation*, 14(8):1771–1800, 2002. doi: 10.1162/089976602760128018.

A. Iserles. On the method of Neumann series for highly oscillatory equations. *BIT Numerical Mathematics*, 44, 2004.

A. Iserles and S. P. Nørsett. On the solution of linear differential equations in lie groups. *Philosophical Transactions: Mathematical, Physical and Engineering Sciences*, 357 (1754):983–1019, 1999. ISSN 1364503X.

Norman L. Johnson, Samuel Kotz, and N. Balakrishnan. *Continuous Univariate Distributions*, volume 1. Wiley, New York, second edition, 1994.

Michael I. Jordan and R. A. Jacobs. Hierarchical mixtures of experts and the EM algorithm. *Neural Computation*, 6:181–214, 1994.

J urgen Jost and Xianqing Li-Jost. *Calculus of Variations.* Cambridge University Press, Cambridge, 1998.

S J Julier and J K Uhlmann. Unscented filtering and nonlinear estimation. *Proc. IEEE*, 92(3):401–422, March 2004.

V. Jurdjevic and H. J Sussmann. Control systems on Lie groups. *Journal of Differential Equations*, 12(2):313–329, 1972. ISSN 0022-0396. doi: https://doi.org/10.1016/0022-0396(72)90035-6.

Y. Y. Kagan. Correlations of earthquake focal mechanisms. *Geophysical Journal International*, 110(2):305–320, 1992. doi: 10.1111/j.1365-246X.1992.tb00876.x.

R. E. Kalman. A new approach to linear filtering and prediction problems. *Transactions of the American Society of Mechanical Engineering, Series D, Journal of Basic Engineering*, 82:35–35, 1960a.

R. E. Kalman. Mathematical description of linear dynamical systems. *Journal of the Society for Industrial and Applied Mathematics Series A Control*, 1:152–192, 1963.

R E Kalman and R S Bucy. New results in linear filtering and prediction theory. *J. Basic Eng*, 83(1):95–108, March 1961.

R.E. Kalman. Contributions to the theory of optimal control, 1960b.

Anatole Katok and Boris Hasselblatt. *Introduction to the Modern Theory of Dynamical Systems.* Cambridge University Press, Cambridge, 1995.

S. Kobayashi and K. Nomizu. *Foundations of Differential Geometry*, volume 1. Wiley Interscience, New York, 1963.

Siem Jan Koopman, André Lucas, and Marcel Scharth. Predicting time-varying parameters with parameter-driven and observation-driven models. *The Review of Economics and Statistics*, 98(1):97–110, 2016. doi: 10.1162/REST_a_00533. URL `https://doi.org/10.1162/REST_a_00533`.

S C Kou, Benjamin P Olding, Martin Lysy, and Jun S Liu. A multiresolution method for parameter estimation of diffusion processes. *Journal of the American Statistical Association*, 107(500):1558–1574, December 2012.

Ryogo Kubo. *A Stochastic Theory of Line Shape*, pages 101–127. Wiley-Blackwell, 2007. ISBN 9780470143605. doi: 10.1002/9780470143605.ch6.

Solomon Kullback. *Information Theory and Statistics*. Dover, New York, 1968.

Arthur D. Kuo, J. Maxwell Donelan, and Andy Ruina. Energetic consequences of walking like an inverted pendulum: Step-to-step transitions. *Exercise and Sport Science Review*, 33(2):88–97, 2005.

Taesoo Kwon and Jessica K. Hodgins. Momentum-mapped inverted pendulum models for controlling dynamic human motions. *ACM Trans. Graph.*, 36(1), January 2017. ISSN 0730-0301. doi: 10.1145/2983616.

Neil D. Lawrence. Gaussian process latent variable models for visualisation of high dimensional data. In *In NIPS*, page 2004, 2003.

Neil D. Lawrence and Joaquin Quiñonero Candela. Local distance preservation in the GP-LVM through back constraints. In *Proceedings of the 23rd International Conference on Machine Learning*, ICML '06, pages 513–520, New York, NY, USA, 2006. ACM. ISBN 1-59593-383-2. doi: 10.1145/1143844.1143909.

Neil D. Lawrence, Guido Sanguinetti, and Magnus Rattray. Modelling transcriptional regulation using Gaussian processes. In *In NIPS*, volume 19, pages 785–792, 2006.

Y. Le Cun, Y. B. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard, and L. D. Jackel. Backpropogation applied to handwritten zip code recognition. *Neural Computation*, 1:541–551, 1989.

Y. Le Cun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86:2278–2324, 1998.

Benn Macdonald, Catherine Higham, and Dirk Husmeier. Controversy in mechanistic modelling with gaussian processes. In Francis Bach and David Blei, editors, *Proceedings of the 32nd International Conference on Machine Learning*, volume 37 of *Proceedings of Machine Learning Research*, pages 1539–1547, Lille, France, 07–09 Jul 2015. PMLR. URL `http://proceedings.mlr.press/v37/macdonald15.html`.

Wilhelm Magnus. On the exponential solution of differential equations for a linear operator. *Communications on Pure and Applied Mathematics*, VII:649–673, 1954.

Kanti V. Mardia and Peter E. Jupp. *Directional Statistics*. Wiley, Chichester, 2000.

J. M. Marin, Pierre Pudlo, Christian P. Robert, and Robin J. Ryder. Approximate Bayesian computational methods. *Statistical Computing*, pages 1167–1180, 2012.

A. W. Matz. Maximum likelihood parameter estimation for the quartic exponential distribution. *Technometrics*, 20(4):475–484, 1978. doi: 10.1080/00401706.1978.10489702.

Kevin McGoff, Sayan Mukherjee, and Natesh Pillai. Statistical inference for dynamical systems: A review. *Statistics Surveys*, 9:209–252, 2015.

Trevelyan McKinley, Alex R. Cook, and Rob Deardon. Inference in epidemic models without likelihoods. *The International Journal of Biostatistics*, 5:24–24, 01 2009. doi: 10.2202/1557-4679.1171.

G. J. McLachlan and T. Krishnan. *The EM Algorithm and its extensions*. Wiley, New York, 1997.

P. Del Moral, A. Doucet, and A. Jastra. An adaptive sequential Monte Carlo method for approximate Bayesian computation. *Statistical Computing*, 22:1009–1020, 2012.

P.A.P. Moran. Quaternions, haar measure and the estimation of a palaeomagnetic rotation. *Perspectives in Probability and Statistics*, 12, 01 1975. doi: 10.1017/S0021900200047720.

A. Morozov and Sh. Shakirov. Introduction to integral discriminants. *Journal of High Energy Physics*, 2009(12):002, 2009.

Franck Multon, Laure France, Marie-Paule Cani-Gascuel, and Giles Debunne. Computer animation of human walking: a survey. *The Journal of Visualization and Computer Animation*, 10(1):39–54, 1999. doi: 10.1002/(SICI)1099-1778(199901/03)10:1⟨39::AID-VIS195⟩3.0.CO;2-2.

Bernt Øksendal. *Stochastic Differential Equations*. Springer, New York, 2010.

Simon Olsson and Frank Noé. Mechanistic models of chemical exchange induced relaxation in protein nmr. *Journal of the American Chemical Society*, 139:200–210, 2017.

Dirk Ormoneit, Hedvig Sidenbladh, Michael J. Black, and Trevor Hastie. Learning and tracking cyclic human motion. In *Advances in Neural Information Processing Systems 13, Papers from Neural Information Processing Systems (NIPS) 2000, Denver, CO, USA*, pages 894–900, 2000.

A. L. O'Toole. A method of determining the constants in the bimodal fourth degree exponential function. *The Annals of Mathematical Statistics*, 4(2):79–93, 05 1933. doi: 10.1214/aoms/1177732802.

Michael J. Prentice. Fitting smooth paths to rotation data. *Journal of the Royal Statistical Society. Series C (Applied Statistics)*, 36(3):325–331, 1987. ISSN 00359254, 14679876.

Xin Qi and Hongyu Zhao. Asymptotic efficiency and finite-sample properties of the generalized profiling estimation of parameters in ordinary differential equations. *The Annals of Statistics*, 38(1):435–481, February 2010.

J. O. Ramsay, G. Hooker, D. Campbell, and J. Cao. Parameter estimation for differential equations: a generalized smoothing approach. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 69(5):741–796, 2007. doi: 10.1111/j.1467-9868.2007.00610.x.

CE. Rasmussen and CKI. Williams. *Gaussian Processes for Machine Learning.* Adaptive Computation and Machine Learning. MIT Press, Cambridge, MA, USA, jan 2006.

H. E. Rauch, C. T. Striebel, and F. Tung. Maximum likelihood estimates of linear dynamic systems. *AIAA Journal*, 3:1445–1450, 1965.

Hannes Risken. *The Fokker-Planck Equation.* Springer-Verlag, Berlin, second edition, 1996.

Sam Roweis and Zoubin Ghahramani. A unifying review of linear Gaussian models. *Neural Computation*, 11(2):305–345, February 1999. ISSN 0899-7667. doi: 10.1162/089976699300016674.

Donald B. Rubin. Bayesianly justifiable and relevant frequency calculations for the applied statistician. *The Annals of Statistics*, 12(4):1151–1172, 12 1984. doi: 10.1214/aos/1176346785.

Simo Säarkä. *Bayesian Filtering and Smoothing.* Cambridge University Press, Cambridge, 2013.

Simo Säarkä and Arno Solin. *Applied Stochastic Differential Equations.* Cambridge University Press, Cambrdige, 2019.

William E. Schiesser. *The Numerical Method of Lines: Integration of Partial Differential Equations.* Academic Press, London, 1991. ISBN 0126241309.

Galit Shmueli. To explain or to predict? *Statistical Science*, 25(3):289–310, 08 2010. doi: 10.1214/10-STS330.

Hedvig Sidenbladh, Michael J. Black, and David J. Fleet. Stochastic tracking of 3D human figures using 2D image motion. In *Proceedings of the 6th European Conference on Computer Vision-Part II*, ECCV '00, pages 702–718, London, UK, UK, 2000. Springer-Verlag. ISBN 3-540-67686-4.

B. W. Silverman. *Density Estimation for Statistics and Data Analysis.* Chapman Hall/CRC, London, 1986.

E. Solak, R. Murray-smith, W. E. Leithead, D. J. Leith, and Carl E. Rasmussen. Derivative observations in Gaussian process models of dynamic systems. In S. Becker, S. Thrun, and K. Obermayer, editors, *Advances in Neural Information Processing Systems 15*, pages 1057–1064. MIT Press, 2003.

Eduardo D. Songtag. *Mathematical Control Theory: Determinstic Finite Dimensional Systems.* Springer, New York, second edition, 1998.

Stan Development Team. Stan modelling language users guide and reference manual, version 2.18.0, 2018.

Charles J. Stone. Consistent nonparameteric regression. *Annals of Statistics*, 5:595–620, 1977.

J.L Strand. Random ordinary differential equations. *Journal of Differential Equations*, 7(3):538 – 553, 1970. ISSN 0022-0396. doi: https://doi.org/10.1016/0022-0396(70)90100-2.

Hector J Sussmann and Velimir Jurdjevic. Controllability of nonlinear systems. *Journal of Differential Equations*, 12(1):95–116, 1972. ISSN 0022-0396. doi: https://doi.org/10.1016/0022-0396(72)90007-1.

R. Deardon T. McKinley, A. R. Cook. Inference in epidemic models without likelihoods. *The International Journal of Biostatistics*, 5:24 – 24, 2009.

Daniel J. Tait and Bruce J. Worton. Multiplicative latent force models. In D. Durante, S. Wade, and Raffaele Argiento, editors, *Bayesian Statistics: New Challenges and New Generations — BAYSM 2018*, page forthcoming. Springer, New York, 2018.

A. Tarantola. *Inverse Problem Theory and Methods for Model Parameter Estimations*. SIAM, 2005.

George Tauchen and Mark Pitts. The price variability-volume relationship on speculative markets. *Econometrica*, 51(2):485–505, 1983. URL `https://EconPapers.repec.org/RePEc:ecm:emetrp:v:51:y:1983:i:2:p:485-505`.

Simon Tavaré, David J. Balding, R. C. Griffiths, and Peter Donnelly. Inferring coalescence times from DNA sequence data. *Genetics*, 145(2):505–518, 1997. ISSN 0016-6731.

Yee Whye Teh, Matthias Seeger, and Michael I. Jordan. Semiparametric latent factor models. In *AISTATS*, 2005.

Michael E. Tipping and Christopher M. Bishop. Probabilistic principal component analysis. *Journal of the Royal Statistical Society. Series B (Statistical Methodology)*, 61(3):611–622, 1999. ISSN 13697412, 14679868.

Michalis Titsias and Neil D. Lawrence. Bayesian Gaussian process latent variable model. In Yee Whye Teh and Mike Titterington, editors, *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*, volume 9 of *Proceedings of Machine Learning Research*, pages 844–851, Chia Laguna Resort, Sardinia, Italy, 13–15 May 2010. PMLR.

Raquel Urtasun, David J. Fleet, Andreas Geiger, Jovan Popović, Trevor J. Darrell, and Neil D. Lawrence. Topologically-constrained latent variable models. In *Proceedings of the 25th International Conference on Machine Learning*, ICML '08, pages 1080–1087, New York, NY, USA, 2008. ACM. ISBN 978-1-60558-205-4. doi: 10.1145/1390156.1390292.

N. G. van Kampen. A cumulant expansion for stochastic linear differential equations. I. *Physica*, 74(2):215–238, 1974. ISSN 0031-8914. doi: https://doi.org/10.1016/0031-8914(74)90121-9.

N. G. van Kampen. *Stochastic Processes in Physics and Chemistry*. Elsevier, third edition, 2007.

J. M. Varah. A spline least squares method for numerical parameter estimation in differential equations. *Journal on Scientific and Statistical Computing*, 3(1):28–46, 1982.

Cédric Villani. *Optimal Transport, Old and New*. Springer, New York, 2008.

V. Vyshemirsky and M. Girolami. Bayesian ranking of biochemical system models. *Bioinformatics*, 24:833–839, 2008.

Jack Wang, Aaron Hertzmann, and David J Fleet. Gaussian process dynamical models. In Y Weiss, B Schölkopf, and J C Platt, editors, *Advances in Neural Information Processing Systems 18*, pages 1441–1448. MIT Press, 2006.

Jack M Wang, David J Fleet, and Aaron Hertzmann. Gaussian process dynamical models for human motion. *IEEE Trans. Pattern Anal. Mach. Intell.*, 30(2):283–298, February 2008.

E. T. Whittaker and Sir William McCrae. *A Treatise on the Analytical Dynamics of Particles and Rigid Bodies*. Cambridge Mathematical Library. Cambridge University Press, Cambridge, fourth edition, 1947. doi: 10.1017/CBO9780511608797.

Christopher K. I. Williams and Carl Edward Rasmussen. Gaussian processes for regression. In *Advances in Neural Information Processing Systems 8*, pages 514–520. MIT press, 1996.

Y. Xie, B. C. Vemuri, and J. Ho. Statistical analysis of tensor fields. In *Medical Image computing and computer-assisted intervention: MICCAI*, volume 13, pages 682–689, 2010.

Hongqi Xue, Hongyu Miao, and Hulin Wu. Sieve estimation of constant and time-varying coefficients in nonlinear ordinary differential equation models by considering both numerical error and measurement error. *The Annals of Statistics*, 38(4):2351–2387, 08 2010. doi: 10.1214/09-AOS784.

Y. Yacoob and M. J. Black. Parameterized modeling and recognition of activities. In *Sixth International Conf. on Computer Vision, ICCV'98*, pages 120–127, Mumbai, India, January 1998.

P. Zarchan and H. Musoff. *Fundamentals of Kalman Filtering: A Practical Approach*. AIAA, second edition, 2005.

P. Zoller, G. Alber, and R. Salvador. ac Stark splitting in intense stochastic driving fields with Gaussian statistics and non-Lorentzian line shape. *Physical Review A*, 24: 398–410, Jul 1981. doi: 10.1103/PhysRevA.24.398.

Appendix A

Some properties of the multivariate Gaussian distribution

The following fundamental properties of multivariate Gaussian distributions are used frequently throughout this work, and so we state the results here so as to provide a convenient reference. We follow the presentation in [Bishop, 2006] where full derivations may be found.

A.1 Coefficient matching for Gaussian distributions

For a Gaussian random variable $\mathcal{N}(\mathbf{z} \mid \boldsymbol{\mu}, \boldsymbol{\Sigma})$ we recall that the log-density is given by

$$\log \mathcal{N}(\mathbf{z} \mid \boldsymbol{\mu}, \Sigma) = -\frac{1}{2}\left(\mathbf{z}^\top \boldsymbol{\Sigma}^{-1}\mathbf{z} - 2\mathbf{z}^\top \boldsymbol{\Sigma}\boldsymbol{\mu}\right) + \text{constant}, \tag{A.1}$$

where the constant does not depend on $\mathbf{z}$. By identifying coefficients we can conclude that if $\mathbf{z}$ is a multivariate random variable supported on R^D with log-density of the form

$$\begin{aligned}\log p(\mathbf{z} \mid \mathbf{A}, \mathbf{b}) &= -\frac{1}{2}\left(\mathbf{zAz} - 2\mathbf{zb}\right) + \text{constant} \\ &= -\frac{1}{2}\left(\mathbf{zAz} - 2\mathbf{zA}^{-1}\mathbf{Ab}\right)\end{aligned} \tag{A.2}$$

where we have assumed that $\mathbf{A}$ is non-singular, then matching the coefficients in (A.1) and (A.2) we conclude that

$$p(\mathbf{z} \mid \mathbf{A}, \mathbf{b}) = \mathcal{N}\left(\mathbf{z} \mid \mathbf{Ab}, \mathbf{A}^{-1}\right). \tag{A.3}$$

A.2 Marginal and conditional Gaussian distributions

Given a marginal Gaussian distribution for the component $\mathbf{x}$ and a conditional Gaussian distribution for $\mathbf{y}$ given $\mathbf{x}$, both taking the form

$$p(\mathbf{x} = \mathcal{N}(\mathbf{x} \mid \boldsymbol{\mu}, \boldsymbol{\Lambda}^{-1}) \tag{A.4}$$

$$p(\mathbf{y} \mid \mathbf{x}) = \mathcal{N}(\mathbf{y} \mid \mathbf{Ax} + \mathbf{b}, \mathbf{L}^{-1}) \tag{A.5}$$

where $\mathbf{A}$ is a rectangular matrix, $\mathbf{b}$ is an offset vector, and $\mathbf{\Gamma}$ and $\mathbf{L}$ are the precision matrices of the marginal and the conditional distributions respectively. Then the marginal distribution of $\mathbf{y}$, and the conditional distribution of $\mathbf{x}$ given $\mathbf{y}$ take the form

$$p(\mathbf{y}) = \mathcal{N}\left(\mathbf{y} \mid \mathbf{A}\boldsymbol{\mu} + \mathbf{b}, \mathbf{L}^{-1} + \mathbf{A}\mathbf{\Lambda}^{-1}\mathbf{A}^{\top}\right) \tag{A.6a}$$

$$p(\mathbf{x} \mid \mathbf{y}) = \mathcal{N}\left(\mathbf{X} \mid \mathbf{\Sigma}\left\{\mathbf{A}^{\top}\mathbf{L}(\mathbf{y} - \mathbf{b}) + \mathbf{\Lambda}\boldsymbol{\mu}\right\}, \mathbf{\Sigma}\right), \tag{A.6b}$$

where the posterior covariance matrix is given by

$$\mathbf{\Sigma} = \left(\mathbf{\Lambda} + \mathbf{A}^{\top}\mathbf{L}\mathbf{A}\right)^{-1}. \tag{A.7}$$

Appendix B

Posterior conditionals for the MLFM-AG model

We provide more exhausive details on the derivation of some of the posterior conditional distributions derived for the MLFM-AG than those presented in the main body of the thesis. For all of the results below it is useful to recall that up to an additive constant we have

$$\begin{aligned}&\log p(\mathbf{X} \mid \boldsymbol{\beta}, \mathbf{g}, \boldsymbol{\phi}, \boldsymbol{\gamma}) \\ &\quad= -\frac{1}{2}\sum_{k=1}^{K} \mathbf{x}_k^\top \mathbf{C}_{\phi_k}^{-1}\mathbf{x}_k - \frac{1}{2}\sum_{k=1}^{K}(\mathbf{f}_k - \mathbf{m}_k)^\top \mathbf{S}_k^{-1}(\mathbf{f}_k - \mathbf{m}_k) + \text{constant}\end{aligned} \tag{B.1}$$

B.1 Posterior condtional for the latent force variables

To construct the conditional distribution for the latent forces conditional on the latent states and additional model parameters we use the representation given by (3.19b). Because of the presence of the offset matrix, $\mathbf{A}_0$, and the fact that the predicted mean $\mathbf{m}_k$ has no functional dependence on the latent forces means that, unlike the case for the latent states, we can not directly expand the argument of the exponential in (3.17) as a homogenous quadratic form. Instead we have

$$\mathbf{f}_k - \mathbf{m}_k = \sum_{r=1}^{R} \mathbf{v}_{kr} \circ \mathbf{g}_r - (\mathbf{m}_k - \mathbf{v}_{k0}),$$

and therefore

$$\begin{aligned}(\mathbf{f}_k - \mathbf{m}_k)^\top \mathbf{S}_k^{-1}(\mathbf{f}_k - \mathbf{m}_k) =& \sum_{r=1}^{R}\sum_{s=1}^{R}(\mathbf{v}_{kr} \circ \mathbf{g}_r)^\top \mathbf{S}_k^{-1}(\mathbf{v}_{kr} \circ \mathbf{g}_s) \\ &- \sum_{r=1}^{R}(\mathbf{v}_{kr} \circ \mathbf{g}_r)^\top \mathbf{S}_k^{-1}(\mathbf{m}_k - \mathbf{v}_{k0}) \\ &- \sum_{s=1}^{R}(\mathbf{m}_k - \mathbf{v}_{k0})^\top \mathbf{S}_k^{-1}(\mathbf{v}_{ks} \circ \mathbf{g}_s) \\ &+ (\mathbf{m}_k - \mathbf{v}_{k0})^\top \mathbf{S}_k^{-1}(\mathbf{m}_k - \mathbf{v}_{k0}).\end{aligned} \tag{B.2}$$

The final term does not depend on $\mathbf{g}$ and so we may ignore it in deriving the conditional density. Rearranging all those terms with an explicit dependance on the latent force variables we have the quadratic expression

$$\mathbf{g}^\top \mathbf{V}_k^\top \mathbf{S}_k^{-1} \mathbf{V}_k \mathbf{g} - 2\mathbf{g}^\top \mathbf{V}_k^\top \mathbf{S}_k^{-1}(\mathbf{m}_k - \mathbf{v}_{k0}) + \text{const.}, \tag{B.3}$$

where we have defined the $N \times NR$ block matrix $\mathbf{V}_k$ by

$$\mathbf{V}_k = \left[\operatorname{diag}(\mathbf{v}_{k1}) | \cdots | \operatorname{diag}(\mathbf{v}_{kR}) \right].$$

After summing over all of the state dimensions, and adding the contribution from the prior $p(\mathbf{g} \mid \boldsymbol{\psi}) = \mathcal{N}(\mathbf{g} \mid \mathbf{0}, \mathbf{C}_\psi)$, we have

$$\begin{aligned}\log p(\mathbf{g} \mid \mathbf{X}, \boldsymbol{\beta}, \boldsymbol{\phi}, \boldsymbol{\gamma}, \boldsymbol{\psi}) &= \log p(\mathbf{X} \mid \boldsymbol{\beta}, \mathbf{g}, \boldsymbol{\phi}, \boldsymbol{\gamma}) + \log p(\mathbf{g} \mid \boldsymbol{\psi}) + \text{const.} \\ &= -\frac{1}{2}\left\{ \mathbf{g}^\top \left(\sum_{k=1}^{K} \mathbf{V}_k^\top \mathbf{S}_k^{-1} \mathbf{V}_k + \mathbf{C}_g^{-1} \right) \mathbf{g} - 2\mathbf{g}^\top \sum_{k=1}^{K} \mathbf{V}_k^\top \mathbf{S}_k^{-1}(\mathbf{m}_k - \mathbf{v}_{k0}) \right\} + \text{const.},\end{aligned} \tag{B.4}$$

and from that we can conclude by identifying terms as in Section A.1 that the posterior conditional of the latent force variable will be a Gaussian with density

$$p(\mathbf{g} \mid \mathbf{X}, \mathbf{B}, \boldsymbol{\phi}, \boldsymbol{\psi}, \boldsymbol{\gamma}) = \mathcal{N}\left(\mathbf{g} \mid \mathbf{m}_g(\mathbf{X}, \mathbf{B}, \boldsymbol{\phi}), \mathbf{K}_g(\mathbf{X}, \mathbf{B}, \boldsymbol{\phi}, \boldsymbol{\psi})\right), \tag{B.5}$$

where the mean and variance parameters dependent on the latent trajectory variables as well as the hyperparameters of the latent force and are given explicitly by

$$\mathbf{m}_g(\mathbf{X}, \mathbf{B}, \boldsymbol{\phi}, \boldsymbol{\psi}, \boldsymbol{\gamma}) = \mathbf{K}_g(\mathbf{X}, \mathbf{B}, \boldsymbol{\phi}, \boldsymbol{\psi})^{-1} \sum_{k=1}^{K} \mathbf{V}_k^\top \mathbf{S}_k^{-1}(\mathbf{m}_k - \mathbf{v}_{k0}) \tag{B.6}$$

$$\mathbf{K}_g(\mathbf{X}, \mathbf{B}, \boldsymbol{\phi}, \boldsymbol{\psi}, \boldsymbol{\gamma}) = \left(\sum_{k=1}^{K} \mathbf{V}_k^\top \mathbf{S}_k^{-1} \mathbf{V}_k + \mathbf{C}_{\boldsymbol{\psi}}^{-1} \right)^{-1}. \tag{B.7}$$

B.2 Posterior conditional for β_{rd}

We this time note that

$$\begin{aligned}&\log p(\mathbf{X} \mid \boldsymbol{\beta}, \mathbf{g}, \boldsymbol{\phi}, \boldsymbol{\gamma}) \\ &\quad = -\frac{1}{2}\sum_{k=1}^{K} \mathbf{x}_k^\top \mathbf{C}_{\phi_k}^{-1} \mathbf{x}_k - \frac{1}{2}\sum_{k=1}^{K} (\mathbf{f}_k - \mathbf{m}_k)^\top \mathbf{S}_k^{-1} (\mathbf{f}_k - \mathbf{m}_k) + \text{constant}\end{aligned} \tag{B.8}$$

depends only on $\boldsymbol{\beta}$ through $\mathbf{f}_k$. We further recall from (3.19c) that we have the expansion

$$\begin{aligned}-\frac{1}{2}\sum_{k=1}^{K} (\mathbf{f}_k - \mathbf{m}_k)^\top \mathbf{S}_k^{-1} (\mathbf{f}_k - \mathbf{m}_k) = -\frac{1}{2}\Bigg(&\boldsymbol{\beta}^\top \sum_{k=1}^{K} \mathbf{W}_k^\top \mathbf{S}^{-1} \mathbf{W}_k \boldsymbol{\beta} \\ &- 2\boldsymbol{\beta}^\top \sum_{k=1}^{K} \mathbf{W}_k^\top \mathbf{S}_k^{-1} \mathbf{m}_k \Bigg) \\ &+ \text{const..}\end{aligned} \tag{B.9}$$

Combining these expressions, and again using the prior $p(\boldsymbol{\beta} \mid \boldsymbol{\zeta})\mathcal{N}(\boldsymbol{\beta} \mid \mathbf{0}, \mathbf{C}_\zeta)$, we have the quadratic expression

$$\begin{aligned}
&\log p(\mathbf{X} \mid \boldsymbol{\beta}, \mathbf{g}, \boldsymbol{\phi}, \boldsymbol{\gamma}, \boldsymbol{\zeta}) p(\boldsymbol{\beta} \mid \boldsymbol{\zeta}) \qquad \text{(B.10)}\\
&\quad = -\frac{1}{2}\left(\boldsymbol{\beta}^\top \left(\sum_{k=1}^{K} \mathbf{W}_k^\top \mathbf{S}_k^{-1} \mathbf{W}_k + \mathbf{C}_\zeta^{-1}\right)\boldsymbol{\beta} - 2\boldsymbol{\beta}^\top \sum_{k=1}^{K} \mathbf{W}_k^\top \mathbf{S}_k^{-1} \mathbf{m}_k\right)\\
&\qquad + \text{constant.} \qquad \text{(B.11)}
\end{aligned}$$

where the constant does not depend on $\boldsymbol{\beta}$, and where $\mathbf{W}_k$ is the $(R+1)D \times N$ matrix where the $(rD+d)$th row with indices $r = 0, 1, \ldots, R$ and $d = 1, \ldots, D$ corresponds to the vector $\mathbf{w}_{krd}$, and $\mathbf{w}_{krd}$ is given by (3.20c).

Applying the coefficient matching results from Section A.1 we conclude that $\boldsymbol{\beta}$ has a Gaussian distribution with mean, $\mathbf{m}_\beta$, and covariance matrix, $\mathbf{C}_\beta$, and we write

$$p(\mathbf{B} \mid \mathbf{X}, \mathbf{g}, \boldsymbol{\phi}, \boldsymbol{\gamma}, \boldsymbol{\zeta}) = \mathcal{N}\left(\boldsymbol{\beta} \mid \mathbf{m}_\beta, \mathbf{K}_\beta\right), \qquad \text{(B.12)}$$

with the parameters given by

$$\mathbf{m}_\beta(\mathbf{X}, \mathbf{g}, \boldsymbol{\phi}, \boldsymbol{\gamma}, \boldsymbol{\zeta}) = \mathbf{K}_\beta \sum_{k=1}^{K} \mathbf{W}_k^\top \mathbf{S}_k^{-1} \mathbf{m}_k, \qquad \text{(B.13a)}$$

$$\mathbf{K}_\beta(\mathbf{X}, \mathbf{g}, \boldsymbol{\phi}, \boldsymbol{\gamma}, \boldsymbol{\zeta}) = \left(\sum_{k=1}^{K} \mathbf{W}_k^\top \mathbf{S}_k^{-1} \mathbf{W}_k + \mathbf{C}_\zeta^{-1}\right)^{-1}. \qquad \text{(B.13b)}$$