UNIVERSITY OF CALIFORNIA
SANTA CRUZ

TEMPORAL DATA MODELS VIA STOCHASTIC PROCESSES

A dissertation submitted in partial satisfaction of the
requirements for the degree of

DOCTOR OF PHILOSOPHY

in

STATISTICAL SCIENCE

by

Rui Meng

June 2020

The Dissertation of Rui Meng
is approved:

Prof. Herbert Lee, Chair

Prof. Abel Rodriguez

Prof. Rajarshi Guhaniyogi

Dr. Priyadip Ray

Quentin Williams
Acting Vice Provost and Dean of Graduate Studies

Table of Contents

List of Figures

List of Tables

Abstract

Temporal Data Models via Stochastic Processes

by

Rui Meng

Temporal data modeling plays a vital role in various research including finance, environmental science and neuroscience. Understanding and interpreting the evolutionary system behind temporal data is of interest. This work mainly emphasizes efficient statistical models on temporal data via stochastic processes. In particular, we focus on statistical modeling via two flexible random processes: Markov processes and Gaussian processes.

The first stage of the research involves a novel hidden Markov model based on Markov jump processes for cervical cancer screening test data. This model is able to model the heterogeneity of both individual and time. We provide an efficient and scalable expectation maximization based inference approach. To the best of our knowledge, our model is the first statistical model that is able to scale to a population-level dataset.

Next, we consider an alternative stochastic process that is widely applied in temporal data modeling, the Gaussian process. Motivated by the kernel perspective of inducing-point based sparse Gaussian processes, we propose a general regularization framework of sparse Gaussian processes and extend it into latent variable models. We specifically consider variational inference under our regularization framework in various settings. We theoretically demonstrate that the variational inference under our regularization can be treated by maximizing a log likelihood lower bound on a corresponding empirical Bayesian model. Our framework is illustrated in various settings throughout both synthetic and real datasets.

Building on our proposed regularization framework, we develop a hierarchical sparse latent Gaussian process model specifically for categorical data and then we extend our model to temporal data via dynamical priors. In particular, we propose efficient variational inference to make it applicable to large datasets. Moreover, our model provides a visualization way to summarize the dynamics of categorical data into a low-dimension manifold.

The fourth project is motivated by electronic health record data and spatially varying coefficient linear coregionalization model. We propose a novel nonstationary multivariate Gaussian process model that allows it to model time dependent smoothness, scale and correlation across different dimensions. One special case of our model is emphasized due to its computational efficiency. It allows an efficient inference via Kronecker algebra. Moreover, we provide both Hamiltonian Monte Carlo inference as a fully Bayesian inference and Maximum a Posteriori based inference as an approximate Bayesian inference. Our posterior inference provides a promising way to understand the relation between cross-correlation of clinical variables and health status, which contributes to early disease detection.

This dissertation is dedicated to my grandmother, Lihua Gu, who always love and support me.

Acknowledgments

First and foremost I express my sincere gratitude to my adviser, Herbert Lee, for the continuous support of my Ph.D. study and related research. His enlightening guidance, everlasting encouragements and enormous patience make this dissertation possible. I owe my gratitude to Dr. Priyadip Ray and Dr. Branden Soper for supporting my three years' research experience in Lawrence Livermore National Laboratory and continuous guidance in my research. In addition to them, I would like to thank the rest of my dissertation committee: Professor Abel Rodriguez and Professor Raj Guhaniyogi for their insightful comments, but also for the hard questions which encourage me to enrich my dissertation.

I would like to thank the faculty of Statistics department at UCSC for their dedication to teaching and mentoring and making our department a friendly and positive place to spend the past several years. I appreciate the valuable statistical courses, especially time series provided by Professor Raquel Prado and nonparametric statistics provided by Professor Athanasios Kottas, that I benefited from.

I would like to thank my fellow doctoral students and friends I met at University of California, Santa Cruz for making there years a remarkable experience.

My deepest gratitude goes to my family. I am grateful to my patients Gaotian and Zhemin, who always encourage me to pursue my goals, to my grandmother Lihua and my aunt Zhewen who are always support me whenever I need help. I am incredibly fortunate to have them. At last, I am grateful to my girlfriend Fan, who always loves and supports me every step of the way.

Chapter 1

Introduction

In this chapter, we discuss motivations in Section 1.1. It includes what kind of data we are interested in, how data look like, how those data inspire our methodologies, and which methodologies we focus on. We display objectives and contributions in Section 1.2. This section contains the overview of the structure of the rest of this dissertation, and summarizes both objectives and contributions for each section.

1.1 Motivations

Modeling the temporal data is of interest in a wide range of applications, from finance and economics to environmental science and neuroscience. Temporal modeling plays a critical role in a variety of research fields. This dissertation mainly focuses on statistical modeling for temporal data.

In the age of big data, large multivariate temporal datasets are readily available in various fields of science. Healthcare, in particular, is becoming increasing crucial since it would help doctors to decide how to best treat patients and make the development of new medicines, procedures and tools possible. Interpretable

statistical modeling in healthcare research is in high demand. Motivated by the importance of healthcare problems, this dissertation would involve two types of healthcare data. One is screening data and the other one is electronic health records. Interpretability in our models is a key feature.

We discuss the screening data in Section 1.1.1 and electronic health records in Section 1.1.2, which directly motivates the novel models in Chapter 2 and Chapter 5 separately. In this dissertation, we are also motivated by the limitation of current models, specifically inducing-point based sparse Gaussian process models. The corresponding motivations discussed in Section 1.1.3 motivate us to propose a regularization structure and extend it to different flexible and efficient models in Chapter 3 and Chapter 4.

1.1.1 Screening Data

The screening data we focus on is a cervical cancer screening test dataset from the Cancer Registry of Norway. The Cancer Registry of Norway has run a national cervical cancer screening program since October 1991, collecting all screening and diagnostic results. Though screening guidelines exist (e.g. a pap smear every three years from age 25 to 69), screening is at the discretion of the individual woman. As a result, the number of screening records and the time between screenings vary considerably between women. Three types of exams are used in the screening program: cytology, histology and human papillomavirus (HPV).

Several features of the dataset make it unique. First, it is because of the size of the dataset. Norway's cervical cancer screening program records cervical exams from the entire target population. Thus the dataset is a true population-level dataset, containing over 1.7 million women with more than 10 million exams recorded from 1992 until the end of 2015. In the current dataset all women above

16 years of age with more than one exam have been included. The maximum number of exams observed was 53 while the median number of exams was 6.

Second, the data have been de-identified and slightly obscured for additional anonymization, the details of which can be found in Ursin et al. (2017). One modification made through this process is that all exam dates are coarsened to the month level. Thus we do not have access to the ordering or timing of multiple screening screening exams occurring within the same calendar month. As a consequence, any model for such data must be able to deal with multiple tests, in both type and quantity, performed at each observation. Though this feature of the dataset is introduced artificially, multiple simultaneous tests are common in medical practice and should be incorporated into any robust model of biomedical diagnostics and/or screening.

Finally, because the data come from a population-level screening program and cancer is a relative rare disease, the resulting dataset is highly skewed towards disease-free observations. As a result, certain exam results are particularly rare. While this can be a common problem in biomedical data, particularly in oncology, the problem is compounded by the stated goal of building more personalized predictive models.

1.1.2 Electronic Health Records

The large-scale collection of electronic health records (EHRs) offers the promise of accelerating clinical research for understanding disease progression and improving predictive modeling of patient clinical outcomes (Cheng et al., 2017; Jung and Shah, 2015). Typically, EHR data consist of rich patient information, including but not limited to, demographic information, vital signs, laboratory results, diagnosis codes, prescriptions and treatments. However, it is extremely challenging

to develop models for EHR data. Contributing to these challenges are data quality, data heterogeneity, complex dependencies across multiple time series, irregular sampling rates, systematically missing data, and statistical nonstationarity (Cheng et al., 2017; Ghassemi et al., 2015; Futoma et al., 2017a).

Despite these challenges, the promise of leveraging EHR data to improve patient outcomes has resulted in an explosive growth of research in the past decade. While the existing literature addresses many of the challenges in modeling EHR data, such as irregular sampling rates (Cheng et al., 2017; Li and Marlin, 2016; Futoma et al., 2017a), missing data (Ghassemi et al., 2015) and the modeling of complex dependencies across multiple streams of clinical data (Cheng et al., 2017; Ghassemi et al., 2015), the inability of stationary models to fits EHR data (Jung and Shah, 2015) has received less attention. In this paper, we propose a novel statistical framework based on multivariate Gaussian processes (GPs) to model both time-dependent smoothness, scale and correlation across different clinical variables in EHR data. We explore both model predictive performance as well as inferred nonstationary correlation patterns across different clinical variables. We considered fully Bayesian inference as well as efficient approximate Bayesian inference. Both inferences provide uncertainty quantification on predictions, such as time-varying length scale parameters and correlations across clinical variables.

While biomedical processes can be both multivariate and nonstationary, models which handle both features have not been explored in the context of EHR data, to the best of our knowledge. Sepsis is a prime example of a disease in which correlated multivariate output and nonstationarity may be critical for early identification. Sepsis has been shown to exhibit highly nonstationary variations in the vitals of patients (Cao et al., 2004) while the cross-correlation of these vitals has been shown to be predictive of early onset (Fairchild et al., 2016). While

both multivariate and nonstationary models have been proposed for EHR data, to the best of our knowledge ours is the first model for EHR data which is both nonstationary and multivariate.

We demonstrate our proposed framework by modeling a large EHR dataset composed of emergency department (ED) hospitalization episodes from Kaiser Permanente (KP). The patients were suspected to have an infection and, in a subset of cases, met the clinical criteria for sepsis (Fohner et al., 2019; Seymour et al., 2016). Sepsis is a life-threatening organ dysfunction arising from a dysregulated host response to infection, affecting at least 30 million patients worldwide and resulting in 5 million deaths each year (Fohner et al., 2019; Klompas and Rhee, 2016).

We apply our proposed approach to jointly model systolic pressure, diastolic pressure, heart rate, pulse pressure and oxygen saturation levels. We demonstrate improved model prediction performance and uncertainty quantification over the state-of-art. Since changes in cross-correlations across vital signs is often an indicator of onset of sepsis (Fairchild et al., 2016), we also explore the inferred cross-correlations across the vitals and their relationship with the hourly LAPS2 scores (LAPS2 is a KP specific measure of acute disease burden and is an indicator of the risk state of a patient) (Escobar et al., 2013).

Gaussian processes have a long history in both spatio-temporal statistical modeling (Luttinen and Ilin, 2009) and in machine learning (Rasmussen and Williams, 2005). With the increasing use of EHR data to improve patient health outcomes, there has been an increased application of Gaussian processes to modeling EHR data. Our use of Gaussian processes is motivated by their flexibility in handling nonstationary and correlated multivariate data, which have been extensively applied to spatio-temporal statistical modeling (Cressie and Wikle, 2011). In this

section we briefly overview the recent literature on the use of Gaussian processes in EHR modeling and point out the main contributions of this paper.

EHR data consist of multiple correlated measurements taken over time. As such, multi-output or multi-task Gaussian processes (MTGPs) have been proposed as appropriate models for EHR data. A MTGP framework for modeling the correlation across multiple physiological time series was first proposed in Dürichen et al. (2014). In Ghassemi et al. (2015) the inferred hyper-parameters from a MTGP model were used as compact latent representations used to predict severity of illness in ICU patients. Online patient state prediction (Cheng et al., 2017) and online patient risk assessment (Alaa et al., 2018) were both proposed via a MTGP framework based on large-scale EHR data sets. Personalized treatment effects were predicted via MTGPs in Alaa and van der Schaar (2017). Online MTGPs were combined with RNN classifiers for early sepsis prediction in hospital patients in Futoma et al. (2017b,a). While each of the above approaches are able to learn a correlation structure both within and between clinical time series, all models are both stationary and homoscedastic.

Because hospitalized patients can go through drastic physiological changes in short periods of time, nonstationary models are needed for EHR data (Cao et al., 2004; Jung and Shah, 2015; Hripcsak et al., 2015). One effect of the biological nonstationarity is highly irregular sampling rates for EHR data. This is due to attending healthcare providers adjusting the sampling rates in response to observable changes in patient state. Nonstationary Gaussian processes have been proposed as means of correcting for these highly irregular sampling rates via time warping in Lasko (2015). While this model does directly model nonstationarity in the clinical time series, it does not directly model correlated multivariate data.

Other than directly modeling EHR data, Gaussian processes have been utilized

in a variety of ways with EHR data. They have been used to smooth and regularize data for blackbox optimizers (Li and Marlin, 2016; Futoma et al., 2017b,a; Lasko et al., 2013), as priors for latent hazard functions in modulated point processes (Lasko, 2014), and as components of hierarchical generative models (Schulam et al., 2015).

The flexibility and expressiveness of Gaussian processes clearly offer a powerful framework for modeling complex EHR data. And while Gaussian process models have been proposed to handle either nonstationarity or correlated multivariate EHR data, to the best of our knowledge there has been no attempt to model both nonstationarity and correlated multivariate EHR data.

1.1.3 Sparse Gaussian process

On the other hand, to deal with large datasets in applications, scalable model and computational efficient models are becoming increasingly in high demand.

One class of computational efficient models is relevant to sparse Gaussian process. Sparse Gaussian processes play a big role in both machine learning and statistics research, because of their flexibility and computational efficiency. Most sparse Gaussian processes are based on a low rank approximation, involving inducing points.

With respect to inducing-point based sparse Gaussian process, how to select the inducing points is an open question. Motivated from the kernel representation, we propose a regularization framework for all inducing-point based sparse Gaussian process models which is allowable to balance the maximizing likelihood of our model and minimizing the distributions between inducing points and inputs. This regularization framework contributes to better model prediction, especially in latent variable models where the inputs are unknown.

Based on the regularization, we also propose a novel statistical model for multivariate categorical temporal data, which is able to model the relation across time and across dimensions. Moreover, it is able the visualize the latent dynamics on a low dimensional manifold.

1.2 Objectives and Contributions

In this dissertation, we explore various aspects of temporal data modeling techniques via stochastic processes including Gaussian processes and Markov jump processes. The main innovative contributions of this dissertation are contained in Chapter 2, 3, 4 and 5, all of which include fundamental literature review.

In Chapter 2, we propose a hierarchical statistical model for cervical cancer screening test data. This model is able to explain both individual heterogeneity and time heterogeneity via a mixture latent model and piece-wise constant intensity matrix processes. Efficient inference procedures are proposed based on Expectation Maximization. Model performance is illustrated by comparing it with recurrent neural network models and naive continuous-time inhomogenuous Markov jump process model on both synthetic data and real cervical cancer screening test data.

In Chapter 3, we propose a regularization framework for different inducing-point based sparse Gaussian process models and extend it into latent variable models. We study the variational inference under our regularization framework and prove it can be treated as maximizing a lower bound of log likelihood in a corresponding empirical Bayesian model. Finally, we discuss our model performance in different hyper-parameter settings and in different datasets.

Chapter 4 inherits the regularization approach in Chapter 3. It proposes a hierarchical statistical model for multivariate categorical data and extends it to

temporal data via dynamical priors. Gaussian processes are introduced to model the correlation across dimensions and time (temporal modeling) and inducing inputs are induced for computation efficiency. We illustrate our model on both synthetic data and a real stock index dataset.

In Chapter 5, motivated by the inherent characteristics of electronic health records, modeling the varying smoothness, scale and correlation of the clinical variables across time is of interested. We propose a flexible nonstationary multivariate Gaussian processes (NMGP) as well as a computationally efficient separable model, which a special case of NMGP. Two inference approaches are proposed including Hamilton Monte Carlo as a fully Bayesian inference and Maximum a Posteriori as an approximate Bayesian inference. Our model is illustrated on both synthetic data and electronic health records provided from Kaiser Permanente.

Chapter 2

Hierarchical Hidden Markov Model

In this chapter, we propose a novel hierarchical hidden Markov model motivated by a large cervical screening test dataset provided by the Cancer Registry of Norway. Generally, our model can apply for any temporal dataset. Our approach models both individual heterogeneity and time heterogeneity and keep the principle of parsimony to make it possible for large datasets. We evaluate model performance by comparing it with other state=-of-the-art models and illustrate that our model has better fitting and prediction results on both synthetic data and real cervical cancer screening test data.

The structure of this chapter is as follows. First, we provide the literature review that is relevant to hidden Markov models and statistical models for longitudinal observation data in Section 2.1. Then we summarize the literature of Markov jump processes including relevant models and corresponding inference in Section 2.2. In Section 2.3, we propose a hierarchical statistical model which is allowable to model both individual heterogeneity and time heterogeneity. The corresponding inference is provided in Section 2.4. Section 2.6 shows experimental

results of both synthetic experiments and a cervical cancer screening test experiment.

2.1 Introduction

Population-based screening programs for identifying undiagnosed individuals have a long history in improving public health. Examples include screening programs for cancer (e.g., cervical, breast, colon), tuberculosis and fetal abnormalities. While the primary objective of such programs is to identify and treat undiagnosed individuals, these cancer screening programs and the population-level, longitudinal datasets associated with them, present many opportunities for the data-driven, computational sciences. In conjunction with modern analytic and computational techniques, such data have the potential to yield novel insights into the natural history of diseases as well as improving the effectiveness of the screening programs.

Hidden Markov Models (HMM) are a standard choice for disease progression modeling for at least three reasons. First, the underlying disease is represented as an unobserved, latent Markov process. Second, noisy measurements of the disease states are efficiently incorporated as conditional probability distributions in the emission mechanism. Third, any modeling assumptions for a particular application are easily incorporated into the transition probability matrix and emission mechanism. However, existing HMMs have several key drawbacks that limit their applicability to real-world datasets, and we propose to address several of those issues herein.

Standard HMMs assume that measurements are regularly sampled at discrete intervals which is often not the case in disease screening programs. Measurements are often irregularly sampled because patients come in for screenings at irregular

intervals, even if regular screening tests are recommended. To deal with irregular sampling, Continuous-Time Hidden Markov Models (CTHMM) are often used since they easily handle samples taken at arbitrary time intervals. CTHMMs have been proposed in many applications such as networks (Wei et al., 2002), medicine (Bureau et al., 2003), seismology (Lu, 2017) and finance (Krishnamurthy et al., 2018).

Health status are of interest in the screening test research. It is crucial to consider a latent model for health status. The HMM is the state of the art approach. Most HMM variants consider only discrete time (Subakan et al., 2014, e.g.,). Continuous time HMMs can handle data at any time stamp and therefore are suitable for irregularly-sampled longitudinal data (Liu et al., 2013; Wang et al., 2014). Liu et al. (2015a) summarize and discuss learning approaches for continuous time HMMs and proposes efficient EM-based learning approaches.

A modified transition matrix is generally modeled by letting the transition distribution depend on a set of observed covariates or exogenous time-series via a multinomial logistic function (Hughes et al., 1999; Paroli and Spezia, 2008). It is an alternative way to model the varying transition matrix. In the continuous-time Markov process transition distribution is derived from the jump Markov process using the Kolmogorov equations. While the parallelization of EM algorithms for hidden Markov models has been studied (Li et al., 2008), to the best of our knowledge, there is no literature on efficient inference for continuous-time, time-inhomogeneous HMMs. In this paper, we propose a scalable EM algorithm for the efficient inference of such models, which is much more efficient than a naïve implementation of a standard EM algorithm.

Because the natural history of many diseases depends heavily on the age of the individual, the time-homogeneous assumption is not valid. For this reason, time-

inhomogeneous HMMs are more appropriate. Although such models have many appealing theoretical properties according to the Kolmogorov equations (Zeifman and Isaacson, 1994), parameter inference is intractable in most non-trivial cases. For this reason, many inference studies of continuous-time, time-inhomogeneous HMMs (CTIHMMs) in the medical domain depend on inefficient microsimulations (Sonnenberg and Robert, 1993; Canfell et al., 2004).

Because of the computational issues, many previous HMM models of disease progression assume that the observations come from a homogeneous population. In large populations, this will typically not be the case. For example, in population-level screening data a large proportion of individuals have benign test results while only a small proportion have abnormal test results. Frailty models are proposed as a common methodology in epidemiological modeling (Yen et al., 2010).

To deal with these difficulties we introduce piece-wise constant transition intensity functions, which allow for tractable parameter inference yet are considerably more flexible in terms of time-inhomogeneity. We then propose a latent structure (i.e., frailty model) to capture unobserved population heterogeneity in terms of disease exposure and susceptibility. Specifically, we propose a new hierarchical hidden Markov model for disease progression in which patients are categorized into classes based on risk levels. Due to the cost of the standard EM algorithm inference, we propose an efficient and scalable EM algorithm combining both soft and hard assignment in the E-step and an auto-differentiation based Limited-memory Broyden-Fletcher-Goldfarb-Shanno (L-BFGS) optimization method in the M-step.

An alternative reason why we prefer the inhomogeneous model is that the homogeneuous continuous-time Markov process assume that arrival time follows an

exponential distribution, which may be too limited and does not match biological processes. Doerr et al. (2013) utilize lognormal distributions to model the inter-arrival time, Mdzinarishvili and Sherman (2010) utilize a non-homogenous Poisson process in the Armitage-Doll model for cancer, where the waiting time follows a Weibull distribution. We aware that semi-Markov model is a generalization of Markov model, which relaxes the Markov assumption by assuming that future states do not only depend on the present but also the past through the time since entry in the present state (Foucher et al., 2005). Specifically, in the continuous time Markov process, the distribution of sojourn time depends on the elapsed time from initial entrance while in the semi-Markov process, that depends on the elapsed time from the previous entrance. When the sojourn time in semi-Markov process are exponentially distributed, this process becomes a homogeneous continuous-time Markov process. Although semi-Markov process is feasible, it is difficult to interpret the biology process. Therefore, we select the inhomogeneous continuous-time Markov model for the screening test data.

Our work is motivated by cervical cancer screening data from the Cancer Registry of Norway, but our model is broadly applicable for longitudinal data (panel data) such as clinical data, survey data, and electronic health record data.

2.2 Markov Jump Processes Models

This section reviews the literature related to Markov jump processes. We first define Markov jump process in Section 2.2.1. Then we discuss its modeling in the homogeneous case and inhomogeneous case in Section 2.2.2 and Section 2.2.3 separately. Next, we treat states are not observable or hidden and model them via Markov jump processes. Then models in the homogeneous case and inhomogenous case are discussed in Section 2.2.4 and Section 2.2.5 respectively.

2.2.1 Markov Jump Process

In order to define the Markov jump process, we give a definition of a continuous-time stochastic process.

Definition 2.2.1 (Continuous-time stochastic process)**.** A continuous-time stochastic process, $(X(t))_{t\geq 0}$ with state space S is a collection of random variables $X(t)$ with values in S.

With an at most countable state space S, the distribution of the stochastic process $(X(t)_{t\geq 0})$ is determined by the probability

$$P(X(t_n) = s_n, \forall n = 1, \dots N)$$

for $0 \leq t_1 < t_2 < \ldots < t_N, s_1, s_2, \ldots, s_N \in S$ and $N \in \mathbb{N}$. We consider countable state space S in the rest of this section.

Due to the complexity of the joint distribution, the continuous-time Markov process is introduced and defined as follow.

Definition 2.2.2 (Continuous-time Markov process)**.** A continuous-time stochastic process $(X(t))_{T\geq 0}$ is a continuous-time Markov process if for all $0 = t_0 < t_1 < \ldots < t_N, s_0, s_1, \dots s_N \in S, N \in \mathbb{N}$,

$$P(X(0) = s_0, X(t_1) = s_1, \dots, X(t_N) = s_N) = p_0(s_0) \prod_{n=1}^{N} p_{s_{n-1},s_n}(t_{n-1}, t_n) \quad (2.1)$$

where $p_0(s_0) = P(X_0 = s_0)$ denotes the initial state probability and $p_{s_{n-1},s_n}(t_{n-1}, t_n) = P(X(t_n) = s_n | X(t_{n-1}) = s_{n-1})$ denotes the conditional probability that X changes from state s_{n-1} at time t_{n-1} to state s_n at time t_n.

The conditional probability $p_{i,j}(s, t)$ has three properties:

- $p_{i,j}(t,t) = \begin{cases} 0 & i \neq j \\ 1 & i = j \end{cases}$.
- $0 \leq p_{i,j}(s,t) \leq 1$.
- $\sum_{j \in S} p_{i,j}(s,t) = 1$.

The homogeneous continuous Markov process assumes the homogeneity in the Markov process. Specifically, it assumes that transition probability $p_{i,j}(t_{n-1}, t_n)$ only depends on the length of time interval (t_{n-1}, t_n). Therefore we denote that

$$p_{i,j}(t_{n-1}, t_n) = p_{i,j}(t_n - t_{n-1})\,. \tag{2.2}$$

Definition (2.2.2) does not define the behavior of stochastic process. It means that the process may change all the time in any small time interval. To deal with this issue, jump processes and regular jump processes are defined.

Definition 2.2.3 (Jump process/regular jump process)**.** A stochastic process $(X(t))_{t \geq 0}$ on an at most countable state space is a jump process if the process $X(t)$ has discrete movements. It is a regular jump process if it is a jump process and it only has finite jumps in any time interval with finite length.

For a Markov process $(X(t))_{t \geq 0}$ with state space S the sequence of transition times $(t'_n)_{n \in \mathbb{N}}$ are the times when $X(t)$ jumps i.e.

$$t'_n = \inf(t \geq t'_{n-1} : X(t) \neq X(t'_{n-1})) \qquad (\text{with } t'_0 = 0 \text{ and } \inf \emptyset = \infty)\,.$$

The time between transition time $t'_{n+1} - t'_n$ is called the holding time or inter-

arrival time. And the embedding process is given by $X_0 = X(0)$ and

$$X_n = \begin{cases} X(t'_n) & \text{if } t'_n < \infty \\ \triangle & \text{if } t'_n = \infty \end{cases}, n \in \mathbb{N} \tag{2.3}$$

where $\triangle$ is an arbitrary element not in S.

In the context of a continuous-time Markov process, the Kolmogorov equations, including Kolmogorov forward equations and Kolmogorov backward equations, are a pair of systems of differential equations that describe the time-evolution of transition probability $p_{i,j}(s,t)$, where $i, j \in S$ and $s < t$ are initial and final time respectively.

According to the Markov property in (2.1), we are going to introduce the Chapman-Kolmogorov equations. It conforms

$$\begin{aligned} p_{i,j}(s,t) &= P(X(t) = j | X(s) = i) \\ &= \sum_{k \in S} P(X(t) = j, X(u) = k | X(s) = i) \\ &= \sum_{k \in S} P(X(t) = j | X(u) = k) p(X(u) = k | X(s) = i) \\ &= \sum_{k \in S} p_{i,k}(s,u) p_{k,j}(u,t) \end{aligned} \tag{2.4}$$

where $s \leq u \leq t$. Then Chapman-Kolmogorov equations can also be seen as a definition of Markov process.

Taking the derivative of Chapman-Kolmogorov equations leads to the Kolmogorov forward equation and Kolmogorov backward equation. Let h be an infinitesimally short time, Then given (2.4),

$$p_{i,j}(s, t+h) = \sum_{k \in S} p_{i,k}(s,t) p_{k,j}(t, t+h)$$

Therefore the derivative of $p_{i,j}(s,t)$ with respect to t is

$$\begin{aligned}\frac{\partial p_{i,j}}{\partial t}(s,t) &= \lim_{h\to 0}\frac{p_{i,j}(s,t+h)-p_{i,j}(s,t)}{h}\\ &= \lim_{h\to 0^+}\frac{\sum_{k\in S} p_{i,k}(s,t)p_{k,j}(t,t+h)-p_{i,j}(s,t)}{h}\\ &= \lim_{h\to 0^+}\frac{\sum_{k\neq j} p_{i,k}(s,t)p_{k,j}(t,t+h)+p_{i,j}(s,t)(p_{j,j}(t,t+h)-1)}{h}\\ &= \sum_{k\neq j} p_{i,k}(s,t)\lim_{h\to 0^+}\frac{p_{k,j}(t,t+h)}{h}+p_{i,j}(s,t)\lim_{h\to 0^+}\frac{p_{j,j}(t,t+h)-1}{h}\,.\end{aligned} \tag{2.5}$$

According to the properties of p in (2.2.1), i.e. $p_{k,j}(t,t)=0$ for $k\neq j$ and $p_{k,j}(t,t)=1$ for $k=j$, we define the transition rate from state k to state j at time t

$$q_{k,j}(t)=\frac{\partial p_{k,j}(t,u)}{\partial u}|_{u=t}=\lim_{h\to 0}\frac{p_{k,j}(t,t+h)-p_{k,j}(t,t)}{h} \begin{cases}\lim_{h\to 0}\frac{p_{k,j}(t,t+h)}{h} & k\neq j\\ \lim_{h\to 0}\frac{p_{k,j}(t,t+h)-1}{h} & k=j\end{cases}. \tag{2.6}$$

Inheriting the property of $p_{i,j}(s,t)$ in (2.2.1), transition rates have two properties:

- $q_{i,j}(t)\geq 0$ for $i\neq j$.
- $\sum_{j\in S} q_{ij}(t)=0$.

Plugging (2.6) into (2.5), we achieve Kolmogorov forward equations.

Definition 2.2.4 (Kolmogorov Forward Equations)**.**

$$\frac{\partial p_{i,j}}{\partial t}(s,t)=\sum_{k\in S} p_{i,k}(s,t)q_{k,j}(t)\,.$$

On the other hand, taking the derivative of $p_{i,j}(s,t)$ with respect to s, we have

$$\begin{aligned}
\frac{\partial p_{i,j}}{\partial s}(s,t) &= \lim_{h\to 0} \frac{p_{i,j}(s+h,t) - p_{i,j}(s,t)}{h} \\
&= \lim_{h\to 0^-} \frac{p_{i,j}(s+h,t) - p_{i,j}(s,t)}{h} \\
&= \lim_{h\to 0^+} \frac{p_{i,j}(s,t) - p_{i,j}(s-h,t)}{h} \\
&= \lim_{h\to 0^+} \frac{p_{i,j}(s,t) - \sum_{k\in S} p_{i,k}(s-h,s)p_{k,j}(s,t)}{h} \\
&= \sum_{k\neq i} p_{k,j}(s,t) \lim_{h\to 0} \frac{p_{i,k}(s-h,s)}{h} + p_{i,j}(s,t) \lim_{h\to 0} \frac{1 - p_{i,i}(s-h,s)}{h} \\
&= -\sum_{k\neq i} p_{k,j}(s,t) \frac{\partial p_{i,k}(u,s)}{\partial u}|_{u=s} - p_{i,j}(s,t) \frac{\partial p_{i,j}(u,s)}{\partial u}|_{u=s} \\
&= -\sum_{k\in S} \left(\frac{\partial p_{i,k}(u,s)}{\partial u}|_{u=s} \right) p_{k,j}(s,t)\,. \qquad (2.7)
\end{aligned}$$

Because of the continuity of transition rate, we have $\lim_{h\to 0} \frac{p_{i,j}(t,t+h)}{h} = \lim_{h\to 0} \frac{p_{i,j}(t+h,t)}{h}$, which causes the equivalence between $\frac{\partial p_{i,j}(t,u)}{\partial u}|_{u=t}$ and $\frac{\partial p_{i,j}(u,t)}{\partial u}|_{u=t}$.

Therefore, plugging the equivalent expression to (2.7), we derive the Kolmogorov backward equations.

Definition 2.2.5 (Kolmogorov Backward Equations)**.**

$$\frac{\partial p_{i,j}}{\partial s}(s,t) = -\sum_{k\in S} q_{i,k}(s) p_{k,j}(s,t)\,.$$

With matrix representation, we rewrite the Kolmogorov forward equations as

$$\frac{\partial P}{\partial t}(s,t) = P(s,t)Q(t) \qquad (2.8)$$

and the Kolmogorov backward equations as

$$\frac{\partial P}{\partial s}(s,t) = -Q(s)P(s,t)\,. \qquad (2.9)$$

From the generative perspective, the transition mechanism is described via transition rate $q_{ij}(t)$.

$$P(X(t+h) = j | x(t) = i) = q_{ij}(t)h + o(h) \qquad i \neq j\,.$$

It assumes transition can only happen at most one time within one infinitesimally short time $(t, t+h)$. Then the mechanism is described by

$$P(X(t+h) = i | x(t) = i) = 1 - \sum_{i \neq j} q_{ij}(t)h + o(h) = 1 + q_{ii}(t)h + o(h)\,. \quad (2.10)$$

Transition rate $q_{ii}(t) = -\sum_{i \neq j} q_{ij}(t)$ is derived from (2.10) and it verifies the property of transition rates $\sum_{k \in S} q_{ik}(t) = 0$.

By solving the Kolmogorov forward equation (2.8), we derive the generative equations

$$P(s,t) = P(s,s)\exp(\int_s^t Q(u)du) = \exp(\int_s^t Q(u)du)$$

for any $0 < s < t$. Solving the Kolmogorov backward equation (2.9) can get the same solution as

$$P(s,t) = \exp(\int_s^t Q(u)du)P(t,t) = \exp(\int_s^t Q(u)du)\,.$$

Let $S_{i,s}$ be a random variable describing the sojourn time or waiting time of X(t) in state i starting at time s. Let us define the distribution of $S_{i,s}$ via $G_{i,s}(t)$:

$$G_{i,s}(t) = P(S_{i,s} \geq t)\,. \quad (2.11)$$

Then by the Markov property,

$$\begin{aligned} G_{i,s}(t+h) &= P(S_{i,s} \geq t, S_{i,s+t} \geq h) \\ &= G_{i,s}(t) G_{i,s+t}(h) \,. \end{aligned}$$

On the other hand, according to the definition of G (2.11) and the properties of Q (2.10),

$$\begin{aligned} G_{i,s+t}(h) &= P(S_{i,s+t} > h) \\ &= P(X(s+t+h) = i | X(s+t) = i) \\ &= 1 + q_{ii}(s+t)h + o(h) \,. \end{aligned}$$

Then the derivative of $G_{i,s}(t)$ is

$$\begin{aligned} G'_{i,s}(t) &= \lim_{h \to 0} \frac{G_{i,s}(t+h) - G_{i,s}(t)}{h} \\ &= q_{ii}(s+t) G_{i,s}(t) \,. \end{aligned}$$

The distribution of $G_{i,s}(t)$ is

$$\begin{aligned} G_{i,s}(t) &= \exp\left(\int_0^t q_{ii}(s+u) du \right) G_{i,s}(0) \\ &= \exp\left(\int_s^{s+t} q_{ii}(u) du \right) . \end{aligned} \tag{2.12}$$

Therefore, the density of $S_{i,s}$ is

$$\begin{aligned} f_{S_{i,s}}(t) &= \frac{d(1 - G_{i,s}(t))}{dt} \\ &= -\exp\left(\int_s^{s+t} q_{ii}(u) du \right) q_{ii}(s+t) \,. \end{aligned} \tag{2.13}$$

The homogeneous Markov process assumes constant transition rate, $Q(t) \equiv Q$ and it has some good properties. In the such case, the distribution of $S_{i,s}(t)$ is independent of starting time s so we denote it as $S_i(t)$. We have

$$P(S_i > t) = \exp(q_{ii}t) \tag{2.14}$$

which implies $S_i \sim \text{Exp}(-q_{ii})$.

Given the properties of Q (2.10), the instant transition probability is

$$\begin{aligned}\lim_{h\to 0} p(X(t+h) = j | X(t) = i, X(t+h) \neq i) &= \frac{p(X(t+h) = j | X(t) = i)}{\sum_{j\neq i} p(X(t+h) = j | X(t) = i)} \\ &= \lim_{h\to 0} \frac{q_{ij}h + o(h)}{\sum_{k\neq i} q_{ik} + o(h)} \\ &= \frac{q_{ij}}{\sum_{k\neq i} q_{ik}}\end{aligned} \tag{2.15}$$

for all $j \neq i$.

A poisson process can be treated as a special case of Markov jump processes, in which it has countable infinite states $S = (0, 1, \ldots)$ and the transition rates are defined as

$$q_{ij} = \begin{cases} \lambda & j = i + 1 \\ -\lambda & j = i \\ 0 & \text{otherwise}\,. \end{cases} \tag{2.16}$$

Moreover, the birth and death process is another special case of Markov jump processes, in which it has countable infinite state $S = (0, 1, \ldots)$ and the transition

rates are defined as

$$q_{ij} = \begin{cases} \lambda & j = i+1 \\ \mu & j = i-1 \\ -\lambda - \mu & j = i \\ 0 & \text{otherwise}\,. \end{cases} \tag{2.17}$$

Next, we introduce the homogeneous Markov model and then introduce the inhomogeneous Markov model. To avoid the complex modeling, we do not model the initial state and assume that it is known, which means we assume that $P(s'_0) = 1$.

2.2.2 Homogeneous Markov Model

Let $\boldsymbol{t'} = (t'_0, t'_1, \ldots t'_{T'})$ refer to the underlying transition timestamps with corresponding state $s'_0, s'_1, \ldots, s'_{T'}$. Let the sojourn time be $\boldsymbol{r'} = (r'_0, r'_1, \ldots, r'_{T'-1})$. In the homogeneous case, the sojourn time follows an exponential distribution (2.14) and conditional transition probability is explicit (2.15), then the complete likelihood (CL) is given by

$$CL = \prod_{i=0}^{T'-1} (q_{s'_i, s'_{i+1}} / q_{s'_i}) q_{s'_i} \exp(-q_{s'_i} r'_i) \tag{2.18}$$

where $q_i = -q_{i,i} = \sum_{j \neq i} q_{i,j}$. In this case, the likelihood depends on all $\{s'_i\}_{i=0,\ldots T'-1}$ and $\{r'_i\}_{i=0,\ldots T'-1}$.

Considering the finite state cases where $|S| < \infty$, we rewrite the complete

likelihood indexed by states, instead of time.

$$CL = \prod_{i \neq j \in S} q_{i,j}^{n_{i,j}} \prod_{i \in S} \exp((-q_i \triangle_i)) \tag{2.19}$$

where $n_{i,j}$ is the number of transitions from state i to state j and $\triangle_i$ is the total sojourn time (waiting time) at state i. Due to the new representation, the likelihood information is summarized by summary statistics $\{n_{i,j}\}_{i \in S, j \in S}$ and $\{\triangle_i\}_{i \in S}$. Then (2.19) is more concise than (2.18).

In practice, underlying transition timestamps $\boldsymbol{t}'$ and corresponding states $\boldsymbol{s}'$ are unknown. Only the irregular sampling timestamp $\boldsymbol{t} = (t_0, t_1, \ldots, t_T)$ and corresponding states $\boldsymbol{s} = (s_0, s_1, \ldots, s_T)$ are observable. We let the sojourn time be $\boldsymbol{r} = (r_0, r_1, \ldots, r_{T-1})$.

EM algorithm is proposed for this homogeneous Markov model in Metzner et al. (2007). The expected complete log likelihood (ECLL) is

$$ECLL = \sum_{i \neq j \in S} E[n_{i,j} | \hat{Q}, \boldsymbol{s}] \log(q_{i,j}) - E[\triangle_i | \hat{Q}, \boldsymbol{s}] q_i \,. \tag{2.20}$$

According to the Markov property, the ECLL can be decomposed as

$$ECLL = \sum_{i \neq j \in S} \sum_{k=0}^{T-1} E[n_{i,j} | \hat{Q}, s_k, s_{k+1}] \log(q_{i,j}) - E[\triangle_i | \hat{Q}, s_k, s_{k+1}] q_i \,.$$

Then in the E-step, we compute the conditional expectation of the number of transitions between state i and state j in each time interval $[t_k, t_{k+1})]$, $E[n_{i,j} | \hat{Q}, s_k, s_{k+1}]$ and the conditional expectation of the waiting time at state u for each time interval $[t_k, t_{k+1})]$, $E[\triangle_i | \hat{Q}, s_k, s_{k+1}]$. Then in the M-step, the

transition rate is optimized

$$\hat{q}_{i,j} = \frac{E[n_{i,j}|\hat{Q}, \boldsymbol{s}]}{E[\triangle_i|\hat{Q}, \boldsymbol{s}]}$$

for $i \neq j$ by maximizing the ECLL.

Therefore, the two conditional expectations $E[n_{i,j}|Q, X(0) = a, X(T) = b]$ and $E[\triangle_i|Q, X(0) = a, X(T) = b]$ are of interest. They are derived in Guttorp and Minin (2018); Hobolth and Jensen (2005); Smyth Gordon (2004).

- Time spent in state i:

$$\begin{aligned}
&E[\triangle_i|Q, X(0) = a, X(T) = b] \\
&= E[\triangle_i \mathbf{1}_b(X(T))|Q, X(0) = a]/P(X(T) = b|X(0) = a) \\
&= E[\int_0^T \triangle_i'(t)dt\mathbf{1}_b(X(T))|Q, X(0) = a]/p_{a,b}(0, T) \\
&= \int_0^T E[\lim_{h\to 0} \frac{\triangle_i(t, t+h)}{h}\mathbf{1}_b(X(T))|Q, X(0) = a]dt/p_{a,b}(0, T) \\
&= \int_0^T \lim_{h\to 0} \frac{1}{h}E[\triangle_i(t, t+h)\mathbf{1}_b(X(T))|Q, X(0) = a]dt/p_{a,b}(0, T) \\
&= \int_0^T \lim_{h\to 0} \frac{1}{h}h(1 - q_i h + o(h)) * p_{a,i}(0, t)p_{i,b}(t+h, T)dt/p_{a,b}(0, T) \\
&= \int_0^T p_{a,i}(0, t)p_{i,b}(t, T)dt/p_{a,b}(0, T)\,. \qquad (2.21)
\end{aligned}$$

- Number of transitions between state i and state j:

$$
\begin{aligned}
&E[n_{i,j}|Q, X(0)=a, X(T)=b] \\
&= E[n_{i,j}\mathbf{1}_b(X(T))|Q, X(0)=a]/P(X(T)=b|X(0)=a) \\
&= E[\int_0^T n'_{i,j}(t)dt\mathbf{1}_b(X(T))|Q, X(0)=a]/p_{a,b}(0,T) \\
&= \int_0^T E[\lim_{h\to 0}\frac{n_{i,j}(t,t+h)}{h}\mathbf{1}_b(X(T))|Q, X(0)=a]dt/p_{a,b}(0,T) \\
&= \int_0^T \lim_{h\to 0}\frac{1}{h}E[n_{i,j}(t,t+h)\mathbf{1}_b(X(T))|Q, X(0)=a]dt/p_{a,b}(0,T) \\
&= \int_0^T \lim_{h\to 0}\frac{1}{h}(q_{ij}h+o(h))p_{a,i}(0,t)p_{j,b}(t+h,T)dt/p_{a,b}(0,T) \\
&= q_{i,j}\int_0^T p_{a,i}(0,t)p_{j,b}(t,T)dt/p_{a,b}(0,T)\,. \qquad (2.22)
\end{aligned}
$$

In the literature, there exist three approaches to compute the conditional expectations above. Metzner et al. (2007) considers a eigen-decomposition of the transition intensity matrix Q and the matrix exponential $P(0,t) = e^{Qt}$ can be simplified. Hobolth and Jensen (2011) introduces a uniformization method in Jenson (1953). Let $u = \max_i(-q_{i,i})$ then we define a transition matrix $R = Q/\mu + I$. It means that $Q = (R-I)\mu$. Using the reparameterization and Taylor expansion, the transition matrix $\exp(Qt)$ can be rewritten as

$$
\begin{aligned}
\exp(Qt) &= \exp(ut(R-I)) \\
&= \sum_{n=0}^{\infty}\frac{(ut)^n}{n!}\exp(-ut)R^n \\
&= \sum_{n=0}^{\infty}R^n\mathrm{Poi}(n;ut)\,. \qquad (2.23)
\end{aligned}
$$

The continuous-time Markov model can also be treated as a discrete-time Markov model with transition matrix R subordinated to a Poisson process with intensity u. Any transition probability from state i to state j with time t is

approximated by

$$\begin{aligned} p_{i,j}(0,t) &= \exp(Qt)_{i,j} \\ &= \sum_{n=0}^{\infty} R^n_{i,j} \text{Poi}(n; ut) \\ &= \sum_{n=0}^{K} R^n_{i,j} \text{Poi}(n; ut) \end{aligned}$$

where K is a truncation level.

Using this similar idea, the conditional expectations are expressed by directly inserting the $\exp(Qt)$ series in to the integral.

$$\begin{aligned} E[\triangle_i \mathbf{1}_b(X(t))|Q, X(0) = a] &= \sum_{n=0}^{\infty} \frac{t}{n+1} \sum_{m=0}^{n} (R^m)_{a,i}(R^{n-m}))_{i,b} \text{Poi}(n; ut) \\ E[n_{i,j} \mathbf{1}_b(X(t))|Q, X(0) = a] &= R_{i,j} \sum_{n=1}^{\infty} \frac{t}{n+1} \sum_{m=0}^{n} (R^m)_{a,i}(R^{n-m}))_{j,b} \text{Poi}(n; ut)\,. \end{aligned}$$

For large values of ut, we have $\text{Poi}(n; ut) \approx \mathcal{N}(n; ut, ut)$ and then the tail of Poisson distribution can be bounded by a cumulative normal distribution. Given this approximation, Tataru and Hobolth (2011) gives a suggested truncation level $K = [4 + 6\sqrt{ut} + ut]$.

The third approach comes from Liu et al. (2015b). They leverage a classic method of Van Loan Van Loan (1978). In Van Loan (1978), it shows that $\int_0^t \exp(Qx) B \exp(Q(t-x))dt = \exp(At)_{(1:n),(n+1):(2n)}$ where n is the dimension of Q and A is constructed as $A = \begin{pmatrix} Q & B \\ 0 & Q \end{pmatrix}$. By setting $B = I(i,j)$, where $I(i,j)$ is the matrix with a 1 in the (i,j)th entry and 0 elsewhere, it is easy to derive $E[\triangle_i|Q, X(0) = a, X(T) = b]$ and $E[n_{i,j}|Q, X(0) = a, X(T) = b]$.

An alternative approach to compute the conditional expectation is to solve an ordinary differential equation (ODE) system in Bladt and Sørensen (2005). They

define the auxiliary functions

$$M^i_{a,b}(t) = E[\triangle_i \mathbf{1}_b(X(t))|Q, X(0) = a]$$
$$F^{i,j}_{a,b}(t) = E[n_{i,j}\mathbf{1}_b(X(t))|Q, X(0) = a]$$

which satisfy systems of ODE. Considering any pair of indexes $i, j \in S$, the vectors $M^i_a(t) = (M^i_{a,1}(t), \ldots, M^i_{a,s}(t))$ and $F^{i,j}_a(t) = (F^{i,j}_{a,1}(t), \ldots, F^{i,j}_{a,s}(t))$ satisfy the two systems of ODEs

$$\frac{d}{dt}M^i_a(t) = M^i_a(t)Q + p_{a,i}(0,t)e_i\,, \qquad M^i_a(0) = 0$$
$$\frac{d}{dt}F^{i,j}_a(t) = F^{i,j}_a(t)Q + q_{i,j}p_{a,i}(0,t)e_j\,, \qquad F^{i,j}_a(0) = 0$$

where e_i and e_j are the i^{th} and j^{th} unit vectors.

Instead of modeling the underlying process, directly modeling the irregular sampling data leads to the likelihood as

$$\begin{aligned} L &= \prod_{i=0}^{T-1} p_{s_i,s_{i+1}}(t_i, t_{i+1}) \\ &= \prod_{i=0}^{T-1} \exp(Qr_i)_{s_i,s_{i+1}}\,. \end{aligned}$$

Next, we introduce the expression of complete likelihood in the inhomogeneous case.

2.2.3 Inhomogeneous Markov Model

Due to the density of sojourn time $S_{i,s}$ (2.13), we have

$$CL = \prod_{i=0}^{T'-1} (q_{s'_i,s'_{i+1}}(t'_{i+1})/q_{s'_i}(t'_{i+1}))f_{S_{s'_i,t'_i}}(r_i) \tag{2.24}$$

where $f_{S_{s'_i,t'_i}}(r'_i) = -\exp\left(\int_{t'_i}^{t'_i+r'_i} q_{s'_i,s'_i}(u)du\right) q_{s'_i,s'_i}(t'_i + r'_i)$.

It is obvious that when all transition rates are constant function $q_{i,j}(t) = q_{i,j}$, (2.24) is equivalent to (2.18) or (2.19).

Through rewriting the complete likelihood indexed by state, we have

$$CL = \prod_{i\neq j\in S} \prod_{t\in\mathcal{A}_{i,j}} (q_{i,j}(t)) \left(\prod_{i\in S} \exp\left(-\int_{t\in\mathcal{B}_i} q_i(t)dt\right)\right) \tag{2.25}$$

where $\mathcal{A}_{i,j}$ is a set of change points from state i to state j and $\mathcal{B}$ is a collection of intervals where process stays in state i. In (2.25), the information of the underlying process is summarized in the change-point collection $\mathcal{A}$ and sojourn time collection $\mathcal{B}$.

Given the observations $\boldsymbol{s}$ on irregular sampling timestamps $\boldsymbol{t}$, we directly model the observable data. The likelihood of data in the inhomogeneous case is

$$\begin{aligned} L &= \prod_{i=0}^{T-1} p_{s_i,s_{i+1}}(t_i, t_{i+1}) \\ &= \prod_{i=0}^{T-1} \exp\left(\int_{t_i}^{t_{i+1}} Q(u)du\right)_{s_i,s_{i+1}} . \end{aligned} \tag{2.26}$$

To infer the transition intensity process, we need to model it first. We proposed a concise way to model the intensity by modeling $q_{ij}(t)$ via a piece-wise constant function. We partition time into I disjoint intervals covering the range of observable time. Then we have a set of disjoint partitions $\mathcal{A} = \{A_i\}_{i=1}^I$. Each transition intensity function q_{ij} for $i \neq j$ is a piece-wise constant function via the defined partition $\mathcal{A}$, denoted by $q_{ij}(t) = \sum_{k=1}^I q_{ijk}\mathbf{1}_{A_k}(t)$, where $\mathbf{1}(\cdot)$ is an indicator function and $q_{ijk} \geq 0$. In this case, the inhomogeneous Markov process is treated as a combination of several continuous-time homogeneous Markov processes, and the transition probability matrix Q is computed as a product of transition probability

matrices with respect to their corresponding partitions.

There are two ways to infer the transition intensities. One is inference based on the underlying process using (2.25) and the other is inference based on observations using (2.26). We prefer the second one because of the concise expression.

Considering the piece-wise constant modeling for the transition intensity, the exponential matrix function in the likelihood (2.26) allows decomposition via partitions, suggesting for any time interval (s, t), without loss of generality, assuming $s \in A_i = [a_i, a_{i+1})$ and $t \in A_j = [a_j, a_{j+1})$ where $i \leq j$, we have

$$\exp(\int_s^t Q(u)du) = \exp\left((a_{i+1} - s)Q_i + \sum_{k=i+1}^{j-1}(a_{k+1} - a_k)Q_k + (t - a_j)Q_j\right). \tag{2.27}$$

After discussing Markov jump process models, we would consider the Hidden Markov jump process models where states are not observable. In other words, Markov jump processes are utilized to model the latent states. Let observable time be $\boldsymbol{t} = (t_0, t_1, \ldots, t_T)$ with corresponding observations $\boldsymbol{y} = (y_0, y_1, \ldots y_T)$ and latent states $\boldsymbol{s} = (s_0, s_1, \ldots, s_T)$. We discuss its relevant models in both the homogeneous case and the inhomogeneous case throughout the following two sections.

2.2.4 Homogeneous Hidden Markov Model

We first introduce the homogeneous case. Assume the underlying transition times are $\boldsymbol{t'} = (t'_0, t'_1, \ldots, t'_{T'})$ with corresponding latent states $\boldsymbol{s'} = (s'_0, s'_1, \ldots, s'_{T'})$. Let $t'_{T'+1} = \infty$. The underlying transition times partition the whole time into $T'+1$ intervals $\{(t_i, t_{i+1}]\}_{i=0,\ldots T'}$. Denote sojourn time as $r'_i = t'_{i+1} - t'_i$ for $t = 0, 1, \ldots$. Assume we have the $T + 1$ irregular observations $\boldsymbol{y} = (y_0, y_1, \ldots, y_T)$ with latent

states $\boldsymbol{s} = (s_0, s_1, \ldots, s_T)$ at time $\boldsymbol{t} = (t_0, t_1, \ldots, t_T)$. Each observed time t_j would be allocated in one of the intervals. Without loss of generality, we assume that $t_j \in [t'_{[j]}, t'_{[j]+1})$. It means that at time t_j, the corresponding state s_j should be consistent with the underlying process such that $s_j = s'_{[j]}$. Therefore, the complete likelihood is

$$CL = \prod_{i=0}^{T'-1} (q_{s'_i, s'_{i+1}} / q_{s'_i}) q_{s'_i} \exp(-q_{s'_i} r'_i) \prod_{j=0}^{T} P(y_j | s_j) \mathbf{1}_{s'_{[j]}}(s_j) \,. \tag{2.28}$$

Because we are interested in the strict positive likelihood, we set $s_j = s'_{[j]}$, implying that only the underlying transition information $\{s'_i\}_{i=0,\ldots,T'}$ and $\{t'_i\}_{i=0,\ldots,T'}$ is inferred.

Using the same reordering tricks in (2.18) and (2.19), the complete likelihood (2.28) is rewritten as

$$CL = \prod_{i \neq j \in S} (q_{i,j})^{n_{i,j}} \prod_{i \in S} \exp(-q_i \triangle_i) \prod_{k=0}^{T} P(y_k | s_k) \tag{2.29}$$

where $n_{i,j}$ is the number of transitions from state i to state j and $\triangle_i$ is the total waiting time at state i.

Given a current estimate of parameter $\hat{Q}$, the expected complete log-likelihood follows the form

$$ECLL = \sum_{i \neq j \in S} E[n_{i,j} | \hat{Q}, \boldsymbol{y}] \log(q_{i,j}) - \sum_{i \in S} E[\triangle_i | \hat{Q}, \boldsymbol{y}] q_i + \sum_{k=0}^{T} E[\log P(y_k | s_k) | \hat{Q}, \boldsymbol{y}] \,. \tag{2.30}$$

Given the ECLL, in the M-step of EM algorithm, the intensity matrix Q is computed by taking the derivative with respect Q and letting it to 0, then it

derives

$$\hat{q}_{ij} = \frac{E[n_{i,j}|\hat{Q}, \boldsymbol{y}]}{E[\triangle_i|\hat{Q}, \boldsymbol{y}]}$$

for $i \neq j \in S$ and $\hat{q}_{i,i} = -\hat{q}_i = -\sum_{j \neq i} \hat{q}_{i,j}$.

Conditional expectations would be computed by marginalizing the latent states as follows:

$$\begin{aligned}
E[n_{i,j}|\hat{Q}, \boldsymbol{y}] &= \sum_{\boldsymbol{s}} p(\boldsymbol{s}|\hat{Q}, Y) E[n_{i,j}|\hat{Q}, \boldsymbol{s}] \\
&= \sum_{\boldsymbol{s}} p(\boldsymbol{s}|\hat{Q}, \boldsymbol{y}) \sum_{k=0}^{T-1} E[n_{i,j}|\hat{Q}, s_k, s_{k+1}] \\
&= \sum_{k=0}^{T-1} \sum_{s_k, s_{k+1}} p(s_k, s_{k+1}|\hat{Q}, \boldsymbol{y}) E[n_{i,j}|\hat{Q}, s_k, s_{k+1}] \\
E[\triangle_i|\hat{Q}, \boldsymbol{y}] &= \sum_{\boldsymbol{s}} p(\boldsymbol{s}|\hat{Q}, \boldsymbol{y}) E[\triangle_i|\hat{Q}, \boldsymbol{s}] \\
&= \sum_{\boldsymbol{s}} p(\boldsymbol{s}|\hat{Q}, \boldsymbol{y}) \sum_{k=0}^{T-1} E[\triangle_i|\hat{Q}, s_k, s_{k+1}] \\
&= \sum_{k=0}^{T-1} \sum_{s_k, s_{k+1}} p(s_k, s_{k+1}|\hat{Q}, \boldsymbol{y}) E[\triangle_i|\hat{Q}, s_k, s_{k+1}]
\end{aligned} \tag{2.31}$$

where the posterior probability of state $p(s_k, s_{k+1}|\hat{Q}, \boldsymbol{y})$ is available by the forward backward algorithm as a Soft method (Rabiner, 1989) or is approximated by the Viterbi algorithm as a Hard method.

Since conditional expectations $E[n_{i,j}|\hat{Q}, s_k, s_{k+1}]$ and $E[\triangle_i|\hat{Q}, s_k, s_{k+1}]$ have already been studied in the homogeneous Markov model, we can succeed to get the estimate of q_{ij} in M-step.

2.2.5 Inhomogeneous Hidden Markov Model

Modeling the underlying process in the inhomogeneous case is complicated, because it does not have concise summary statistics like the homogeneous case. Therefore, we directly model the observations and then the likelihood is

$$L = \prod_{i=0}^{T-1} p_{s_i,s_{i+1}}(t_i, t_{i+1}) \prod_{i=0}^{T} p(y_i|s_i)\,. \tag{2.32}$$

To infer the likelihood 2.32, we utilize the EM algorithm. Treating the latent states $\boldsymbol{s}$ as latent variables, given estimates of transition intensities $\hat{\boldsymbol{Q}} = \{\hat{Q}_i\}_{i=1,2,\ldots,I}$, the expected conditional log likelihood is

$$ECLL = E[\log(p(\boldsymbol{s}))|\hat{\boldsymbol{Q}}, \boldsymbol{y}] + E[\log p(\boldsymbol{y}|\boldsymbol{s})|\hat{\boldsymbol{Q}}, \boldsymbol{y}]\,. \tag{2.33}$$

In the E-step, to compute the two posterior expectations above, we need to compute the posterior distribution of $p(\boldsymbol{s}|\hat{\boldsymbol{Q}}, \boldsymbol{y})$. There exist two approaches, one is to compute the posterior using the forward-backward method (Rabiner, 1989) and the other is to approximate the posteriors via using the MAP of states via Viterbi algorithm.

With the forward-backward method, considering the homogeneous discrete Markov model, the EM algorithm is called the Baum-Welch algorithm (Bishop, 2006), which has a closed expression with respect to Q.

2.3 Model

We propose a hierarchical inhomogeneous HMM (HIHMM) to model disease progression. The hierarchical graphical representation is illustrated in Figure 2.1.

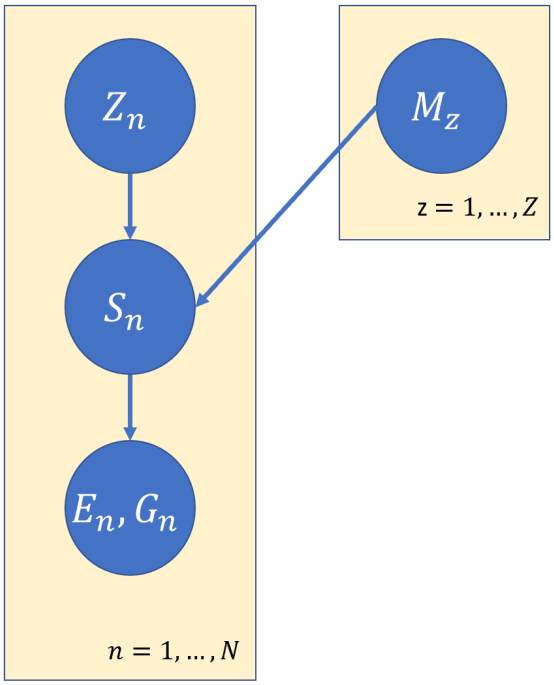

Figure 2.1: Graphical representation for hierarchical inhomgeneous hidden Markov models.

In our real world case study, we assume all patients come from two risk categories: high disease exposure risk and low disease exposure risk. Each category has its own Markov transition structure shown in Figure 2.2 but shares the same emission mechanism. Details are discussed in Section 2.6. This hierarchical structure of the HIHMM allows for an arbitrary number of latent frailty states, provided relevant Markov models can be ascribed to the disease progression associated with each.

2.3.1 Variables

Suppose there are N individuals in the screening population. Let individual n have T_n screening visits at ages $a_1, \ldots, a_{T_n}$. We assume Z categories are considered

in the hierarchical frailty structure and introduce the following variables:

$$\begin{aligned}
&\text{Frailty State (hidden): } z_n \in \{1, \ldots, Z\}\,, \\
&\text{Disease States (hidden): } S_{nt} \in \{1, \ldots, M_{z_n}\}\,, \\
&\text{Number of screening tests (observable): } E_{ntk} \in \mathbb{N}\,, \\
&\text{Screening test results (observable): } G_{ntk} \in \mathbb{N}^{L_k}\,.
\end{aligned}$$

The underlying disease state of individual n is assumed to evolve according to a continuous-time, time-inhomogeneous Markov process assigned by its latent frailty class indicator z_n, where only screening results at specific time stamps with corresponding ages $a_1, \ldots a_{T_n}$ are observable. On the t^{th} screening visit of individual n, S_{nt} refers to the latent disease state and the visit includes E_{ntk} tests of the k^{th} test type. In the screening dataset, E_{ntk} may be greater than 1, because a screening visit is recorded only monthly for patient confidentiality. Corresponding results G_{ntk} is a L_k dimensional vector and the value on the l^{th} dimension refers to the number of the l^{th} grade results.

2.3.2 Model of Disease Progression

As for the z^{th} underlying Markovian disease process, it is parameterized by an $M_z \times M_z$ transition intensity matrix Q_z. For the simplicity of notation, we ignore the subscript z in the remainder of this section. The element q_{ij} in the matrix Q satisfies $q_{ij} \geq 0$ for $i \neq j$ and $q_{ii} = -\sum_{i \neq j} q_{ij}$. The time spent in state i is exponentially distributed with rate $-q_{ii}$. Given that a transition occurs from state i, the probability of transitioning to state j is $\frac{q_{ij}}{q_i}$ where $q_i = \sum_{i \neq j} q_{ij}$. When Q is invariant for time t the model is homogeneous, otherwise the model is inhomegeneous.

Homogeneous Markov Model

For a homogeneous Markov process, assume the initial state at t_1 is known, $p(S(t_1)) = 1$. We let $t'_1 = t_1$ and $\boldsymbol{t'} = (t'_1, \ldots, t'_{T'})$ refer to the underlying transition timestamps and let $\boldsymbol{O} = (O_1, \ldots, O_T)$ denote observations at time $\boldsymbol{t} = (t_1, \ldots, t_T)$. Then the complete likelihood (CL) is:

$$\begin{aligned}\text{CL} &= \prod_{i=1}^{T'} (q_{S(t'_i),S(t'_{i+1})}/q_{S(t'_i)}) q_{S(t'_i)} e^{-q_{S(t'_i)}\triangle_i} \prod_{j=1}^{T} p(O_j|S(t_j)) \\ &= \prod_{i=1}^{M} \left(e^{-q_i \tau_i} \prod_{j \neq i} q_{ij}^{n_{ij}} \right) \prod_{j=1}^{T} p(O_j|S(t_j)) \,,\end{aligned}$$

where $\tilde{\triangle}_i = \tilde{t}_{i+1} - \tilde{t}_i$ and n_{ij} denotes the number of times the state changes from state i to state j during the whole process and τ_i denotes the duration that the process stays in state i. Since the underlying transition timestamps $\boldsymbol{t'}$ are not observable, the marginalized complete likelihood (MCL) is derived by marginalizing all $\boldsymbol{t'}$ as

$$\text{MCL} = \prod_{i=1}^{T-1} P(\triangle_i)_{S(t_i),S(t_{i+1})} \prod_{j=1}^{T} p(O_j|S(t_j)) \,,$$

where $\triangle_i = t_{i+1} - t_i$ and $P(\triangle_i) = e^{Q\triangle_i}$ is the transition probability matrix from time t_i to time t_{i+1}.

Inhomogeneous Markov Model

An inhomogeneous Markov process drops the time invariance assumption of Q, by allowing it to be a function of t. But it requires three conditions:

$$q_{ij}(t) = \begin{cases} -\sum_{i \neq k} q_{ik}(t) & i = j \\ q_{ij}(t) > 0 & i \neq j, i \sim j \\ 0 & i \neq j, i \not\sim j \end{cases} \tag{2.34}$$

where $i \sim j$ means state i connects to state j while $i \not\sim j$ means state i does not connect to state j in the transition structure of model. The CL then becomes intractable, because the time spent in state i no longer follows an exponential distribution. An alternative approach is to consider the MCL. The only difference in the expression of MCLs between homogeneity and inhomogeneity is the computation of the transition matrix $P([t_i, t_{i+1}])$ from time t_i to time t_{i+1} for $i = 1, \ldots, T-1$. For the inhomogeneous model, $P([t_i, t_{i+1}]) = \exp\{\int_{t_i}^{t_{i+1}} Q(t)dt\}$.

The transition intensity function $Q(t)$ can be modeled by any parametric model, but the computation of the matrix exponential $\exp\{\int_{t_i}^{t_{i+1}} Q(t)dt\}$ may be prohibitively expensive, even taking advantage of numerical computational methods. To ease this computational burden, we propose a piecewise constant transition intensity matrix Q. Due to conditions of the model (2.34), we are interested in modeling q_{ij} for $i \sim j$ as a piecewise positive constant function of time. Specifically, we partition time into I disjoint intervals covering the range of observable time. We then have a set of disjoint partitions $\mathcal{A} = \{A_i\}_{i=1}^{I}$. Each transition intensity function q_{ij} is a piecewise constant function via the defined partition $\mathcal{A}$, denoted by $q_{ij}(t) = \sum_{k=1}^{I} q_{ijk}\mathbf{1}_{A_k}(t)$, where $\mathbf{1}(\cdot)$ is an indicator function and $q_{ijk} \geq 0$, for $k = 1, \ldots, I$. In this case, the inhomogeneous Markov process can be treated as a combination of several continuous-time homogeneous Markov processes, and the transition probability matrix Q can be computed as a product of transition probability matrices with respect to their corresponding partitions.

Treatment Modeling

Treatment information of patients is an important factor in disease progression, giving information about how many times and when the patients get treatments. In our case, we model the treatment information as a reset in the transition

structure. Once a patient is treated at certain time, we assume his or her state would automatically transit to the Normal state. For instance, we model the treatment as a dashed line in Figure 2.2.

2.3.3 Hierarchical Model

Due to the significant population heterogeneity related to disease exposure risk, we propose a hierarchical model as follows. Let $\boldsymbol{\psi} = (\boldsymbol{\psi}_1, \ldots, \boldsymbol{\psi}_Z)$ denote all model parameters and $\boldsymbol{\psi}_z$ be parameters for model z. Then the hierarchical model is given by

$$\begin{aligned} \boldsymbol{O}_n &\sim \mathcal{M}_{z_n}(\boldsymbol{\psi}_{z_n}, \boldsymbol{\theta}_n)\,, \\ z_n &\sim \mathrm{Cat}(\boldsymbol{p})\,, \end{aligned}$$

where $\boldsymbol{\theta}_n$ denotes all covariates for individual n. An informative prior of the model indicator z_n is proposed as a categorical distribution with hyper-parameters $\boldsymbol{p}$, which is used to provide expert knowledge of the model assignment. This prior contributes to reasonable model inference, especially when screening data are highly unbalanced in terms of latent class membership. Figure 2.2 shows the case where $Z = 2$ and index z_n has a Bernoulli prior with a parameter p, i.e., $z_n \sim \mathrm{Ber}(p)$.

2.4 Inference

This section gives a scale expectation maximization approach for inference. We also provide inference procedures for treatment information.

2.4.1 Scalable Expectation Maximization Approach

Due to the latent characteristics of both model indices and patient states, the expectation maximization (EM) approach is considered. Our EM algorithm employs the true conditional posterior for the model index and the pseudo-conditional posterior for states in the E-step. Both hard-assignment and soft assignment approaches are studied in the literature but they are not combined. We balance the advantages of both methods for inference. Specifically, considering the heterogeneity of our model, we marginalize the latent transition timestamps in our inference. We decompose the joint posterior distribution as $p(z_n, \boldsymbol{S}_n|-) = p(z_n|-)p(\boldsymbol{S}_n|z_n, -)$, where $-$ denotes all other parameters and use soft assignment for $p(z_n|-)$ and hard assignment for $p(\boldsymbol{S}_n|z_n, -)$, Because the computation of all possible $p(\boldsymbol{S}_n|z_n, -)$ is prohibitively expensive. Specifically, the number of possible results $\boldsymbol{S}_n$ is $M_{z_n}^{T_n}$. Calculating the joint probability of each state sequence with the observed series event costs $O(T_n M_{z_n}^{T_n})$. When the length of time series T_n is large, the computation is infeasible. We are aware that the hard assignment with Viterbi algorithm may affect the algorithm to get a different local mode. Although some sensitivity of result comes from Viterbi approximation, the largest source of sensitivity of the local mode comes from the initial point selection. The sensitivity of the Viterbi approximation is likely to be small relative to the sensitivity to initial point selection. We also note that Monte Carlo approximation is an alternative way to approximate the $p(\boldsymbol{S}_n|z_n, -)$. But considering the high efficiency of our algorithm, we select the hard assignment approach. For simplicity, we ignore covariates $\boldsymbol{\theta}_n$ in the remainder of this section.

The recursive procedures are given as follows:

- Given previous estimates $\boldsymbol{\psi}^{(t-1)}$, compute the conditional posterior distri-

bution of z_n:

$$p(z_n|\boldsymbol{O}_n, \boldsymbol{\psi}^{(t-1)}) \propto \pi(z_n) p(\boldsymbol{O}_n|z_n, \boldsymbol{\psi}^{(t-1)}) \\ \sim \text{Cat}(\tilde{\boldsymbol{p}}_n)\,, \tag{2.35}$$

where $\tilde{p}_{nk} = \frac{p_k p(\boldsymbol{O}_n|z_n=k, \boldsymbol{\psi}_k^{(t-1)})}{\sum_{z=1}^{Z} p_z p(\boldsymbol{O}_n|z_n=z, \boldsymbol{\psi}_z^{(t-1)})}$ for $k = 1, \ldots, Z$ and $p(\boldsymbol{O}_n|z, \boldsymbol{\psi}_z)$ is accessible through the forward-filter backward-sample algorithm (FFBS), which is a sequential Monte Carlo approach first proposed in Kitagawa (1987).

- Update the optimal state sequence $\boldsymbol{S}_n$ given corresponding observations $\boldsymbol{O}_n$ and model indicator z using the Viterbi algorithm (Forney, 1973):

$$\boldsymbol{S}_{nz}^{(t)} = \text{Viterbi}(\boldsymbol{O}_n, \boldsymbol{\psi}_z^{t-1})\,. \tag{2.36}$$

- Maximize the expected marginal complete log-likelihood (EMCLL) with respect to $\boldsymbol{\psi}$ by

$$\begin{aligned} \boldsymbol{\psi}^{(t)} &= \underset{\boldsymbol{\psi}}{\text{argmax}} \sum_{n=1}^{N} E_{z_n, \boldsymbol{S}_n}(\ell(\boldsymbol{\psi}|\boldsymbol{O}_n, z_n, \boldsymbol{S}_n)|\boldsymbol{O}_n, \boldsymbol{\psi}^{(t-1)}) \\ &= \underset{\boldsymbol{\psi}}{\text{argmax}} \sum_{n=1}^{N} \sum_{z=1}^{Z} \sum_{\boldsymbol{S}_n} p(z_n = z|\boldsymbol{O}_n, \boldsymbol{\psi}^{(t-1)}) \\ &\qquad q(\boldsymbol{S}_n|z_n = z, \boldsymbol{O}_n, \boldsymbol{\psi}^{(t-1)}) \ell(\boldsymbol{\psi}|\boldsymbol{O}_n, z, \boldsymbol{S}_n) \\ &= \underset{\boldsymbol{\psi}}{\text{argmax}} \sum_{n=1}^{N} \sum_{z=1}^{Z} p(z_n = z|\boldsymbol{O}_n, \boldsymbol{\psi}^{(t-1)}) \\ &\qquad \Big(\log p_z + \log p(\boldsymbol{S}_{nz}^{(t)}|z, \boldsymbol{\psi}) + \log p(\boldsymbol{O}_n|\boldsymbol{S}_{nz}^{(t)}, \boldsymbol{\psi})\Big)\,. \end{aligned} \tag{2.37}$$

Moreover, we decompose parameters $\boldsymbol{\psi}$ into two parts, transition parameters $\boldsymbol{\psi}^{tran}$ and emission parameters $\boldsymbol{\psi}^{emis}$. Denote $\boldsymbol{\psi}_z^{tran}$ as transition parameters in

the z^{th} model. Due to (2.37), we can separately estimate $\boldsymbol{\psi}_z^{tran}$ and $\boldsymbol{\psi}^{emis}$ by

$$\begin{aligned}
\hat{\boldsymbol{\psi}}_z^{tran(t)} &= \text{argmax} \sum_{n=1}^{N} p(z_n = z|\boldsymbol{O}_n, \boldsymbol{\psi}^{(t-1)}) \log(\boldsymbol{S}_{nz}^{(t)}|z, \boldsymbol{\psi}_z^{tran}) \\
\hat{\boldsymbol{\psi}}_z^{emis(t)} &= \text{argmax} \sum_{n=1}^{N} p(z_n = z|\boldsymbol{O}_n, \boldsymbol{\psi}^{(t-1)}) \log(\boldsymbol{O}_n|\boldsymbol{S}_{nz}^{(t)}, \boldsymbol{\psi}_z^{emis}) \,. \qquad (2.38)
\end{aligned}$$

For a simple emission mechanism, the estimate of ψ_z^{emis} may have a closed-form expression. But there is no closed-form for transition parameters. Numerical optimization is necessary in the M-step.

On the other hand, since population screening datasets contain millions of records, direct inference may be prohibitively expensive and more scalable approaches are necessary. Since the computation complexity of (2.35), (2.36) and (2.38) is linear with respect to the number of observations N, we scale our EM algorithm by parallelizing the inference across observations using $\tilde{N}$ clusters, $\{C_n\}_{n=1}^{\tilde{N}}$, in three parts. We first compute the conditional posterior distribution of z_n in each cluster using (2.35). The time complexity for each cluster is $O(|C_n|ZM^2T)$. We then compute the optimal state sequences in each cluster C_n using (2.36) with the same time complexity $O(|C_n|ZM^2T)$. Finally, we compute the gradients in each cluster then reduce all local gradients to global gradients for the optimization in the M-step.

Automatic differentiation (AD) (Maclaurin, 2016) is utilized to compute the gradients in each cluster. Summing over all clusters, we get the gradient of the EMCLL. Using this gradient we adapt the Limited-Memory BFGS (Liu and Nocedal, 1989) algorithm to estimate $\boldsymbol{\psi}$. Analytically computing the complexity of the L-BFGS algorithm for each cluster is intractable, but the parallelized algorithm increases inference speed around $\tilde{N}$ times. We also note that the log scale transform trick is employed in the L-BFGS when dealing with positive constraint.

2.4.2 Inference with Treatment Information

When considering the treatment information, the disease process is decomposed as several independent disease processes via the treatment timestamps. The inference has three modifications with respect to FFBS, Viterbi and EMCLL respectively.

Without loss of generality we suppose one woman has m treatments indexed by $\{r_1, \ldots, r_m\}$. Then the observation sequence $\boldsymbol{O}$ is partitioned as

$$\{\boldsymbol{O}_1, \ldots, \boldsymbol{O}_{r_1}\}, \ldots, \{\boldsymbol{O}_{r_m}, \ldots \boldsymbol{O}_T, \boldsymbol{O}_c\}.$$

Throughout the FFBS, the marginal likelihood is decomposed as

$$\begin{aligned} p(\boldsymbol{O}|z, \boldsymbol{\psi}) =& p(\boldsymbol{O}_1, \ldots, \boldsymbol{O}_{r_1}|z, \boldsymbol{\psi}) \\ & \prod_{j=1}^{m-1} p(\boldsymbol{O}_{r_j+1}, \ldots, \boldsymbol{O}_{r_{j+1}}|S_{r_j} = 0, z, \boldsymbol{\psi}) \\ & p(\boldsymbol{O}_{r_m+1}, \ldots \boldsymbol{O}_T, \boldsymbol{O}_c|S_{r_m} = 0, z, \boldsymbol{\psi}). \end{aligned} \tag{2.39}$$

Each component of (2.39) is tractable using FFBS (Kitagawa, 1987).

A similar decomposition is implemented in the Viterbi algorithm to find the most likely sequence of hidden states.

$$\begin{aligned} (S_1, \ldots, S_{r_1}) &= \text{Viterbi}(\boldsymbol{O}_1, \ldots, \boldsymbol{O}_{r_1}, \boldsymbol{\psi}) \,, \\ (S_{r_j+1}, \ldots, S_{r_{j+1}}) &= \text{Viterbi}(\boldsymbol{O}_{r_j+1}, \ldots, \boldsymbol{O}_{r_{j+1}}, \boldsymbol{\psi}|S_{r_j}=0) \\ & j = 1, \ldots, m-1 \,, \\ (S_{r_m+1}, \ldots, S_T) &= \text{Viterbi}(\boldsymbol{O}_{r_m+1}, \ldots, \boldsymbol{O}_T, \boldsymbol{O}_c, \boldsymbol{\psi}|S_{r_m}=0) \,. \end{aligned}$$

According to (4), it is sufficient to compute EMCLL by computing $p(\boldsymbol{S}|z, \boldsymbol{\psi})$.

Using a similar decomposition we arrive at $p(\boldsymbol{S}|z,\boldsymbol{\psi}) = p(S_1|z,\boldsymbol{\psi})\prod_{i\in\{r_j\}} p(S_{i+1}|S_i = 0)\prod_{i\notin\{r_j\},i\neq 1} p(S_{i+1}|S_i)$.

2.5 Convergence of proposed EM algorithm

This section gives a theoretical proof of the convergence of our proposed EM algorithm. In general, we denote disease states as $\boldsymbol{S}$, frailty states as $\boldsymbol{z}$, all model parameters as $\boldsymbol{\theta}$ and data as $\boldsymbol{O}$. Our iterative EM algorithm can be summarized as follows:

1 Calculating $p(\boldsymbol{z}|\boldsymbol{\theta}^{(t)},\boldsymbol{O})$

2 Finding the optimal $\boldsymbol{S}_z^{(t)}$ given $\boldsymbol{z}$ that maximize $p(\boldsymbol{S}|\boldsymbol{z},\boldsymbol{\theta}^{(t)},\boldsymbol{O})$.

3 Finding the optimal $\boldsymbol{\theta}^{t+1}$ that maximize $\mathrm{E}_{q(\boldsymbol{z},\boldsymbol{S}|\boldsymbol{\theta}^{(t)})}\log p(\boldsymbol{\theta},\boldsymbol{z},\boldsymbol{S},\boldsymbol{O})$ where $q(\boldsymbol{z},\boldsymbol{S}|\boldsymbol{\theta}^t,\boldsymbol{O}) = p(\boldsymbol{z}|\boldsymbol{\theta}^t,\boldsymbol{O})\mathbf{1}_{\boldsymbol{S}_z^{(t)}}(\boldsymbol{S})$

We recursively repeat the step 1 to step 3 until $\boldsymbol{\theta}$ converges. Next, we are going to prove that $\boldsymbol{\theta}$ must converge to a fixed point $\boldsymbol{\theta}^*$.

Due to Jensen's inequality, the likelihood has the lower bound

$$\ell(\boldsymbol{\theta}) = \log p(\boldsymbol{O}|\boldsymbol{\theta}) \geq E_{q(\boldsymbol{z},\boldsymbol{S})}\log\frac{p(\boldsymbol{O},\boldsymbol{z},\boldsymbol{S},\boldsymbol{\theta})}{q(\boldsymbol{z},\boldsymbol{S})} = \ell^*(\boldsymbol{\theta}) \tag{2.40}$$

for any distribution $q(\boldsymbol{z},\boldsymbol{S})$. We select a structured family of distributions such that $q(\boldsymbol{z},\boldsymbol{S}) = q(\boldsymbol{z})q(\boldsymbol{S}|\boldsymbol{z}) = q(\boldsymbol{z})\mathbf{1}_{\tilde{\boldsymbol{S}}}(\boldsymbol{S}|\boldsymbol{z})$. For any θ, the optimal distribution of $\boldsymbol{z},\boldsymbol{S}$ which maximizes (2.40) is $\tilde{q}(\boldsymbol{z},\boldsymbol{S}) = p(\boldsymbol{z}|\boldsymbol{O},\boldsymbol{\theta})\mathbf{1}_{\arg\max_{\boldsymbol{S}} p(\boldsymbol{S}|\boldsymbol{z},\boldsymbol{O},\boldsymbol{\theta})}(\boldsymbol{S})$ based on the Euler-Lagrange equation.

At the t^{th} iteration, we denote the optimal distribution with respect to $\boldsymbol{\theta}^{(t)}$ as $\tilde{q}^{(t)}(\boldsymbol{z},\boldsymbol{S})$. Then the best lower bound of log likelihood $\tilde{\ell}^{(t)}$ is increasing as the

number of iteration t increases.

$$\begin{aligned}
\ell^*(\boldsymbol{\theta}^{(t+1)}) &= E_{\tilde{q}^{(t+1)}(\boldsymbol{z},\boldsymbol{S})} \log \frac{p(\boldsymbol{O},\boldsymbol{z},\boldsymbol{S},\boldsymbol{\theta}^{(t+1)})}{\tilde{q}^{(t+1)}(\boldsymbol{z},\boldsymbol{S})} \\
&\geq E_{\tilde{q}^{(t)}(\boldsymbol{z},\boldsymbol{S})} \log \frac{p(\boldsymbol{O},\boldsymbol{z},\boldsymbol{S},\boldsymbol{\theta}^{(t+1)})}{\tilde{q}^{(t)}(\boldsymbol{z},\boldsymbol{S})} \qquad (2.41) \\
&\geq E_{\tilde{q}^{(t)}(\boldsymbol{z},\boldsymbol{S})} \log \frac{p(\boldsymbol{O},\boldsymbol{z},\boldsymbol{S},\boldsymbol{\theta}^{(t)})}{\tilde{q}^{(t)}(\boldsymbol{z},\boldsymbol{S})} = \ell^*(\boldsymbol{\theta}) \qquad (2.42)
\end{aligned}$$

(2.41) exists because $\tilde{q}^{(t+1)}$ is the optimal distribution which maximizes the lower bound in (2.40) with respect to $\boldsymbol{\theta}^{(t+1)}$. and finding the $\tilde{q}^{(t+1)}$ is Step 1 and Step 2 in the $(t+1)^{\text{th}}$ iteration. (2.42) exists because of Step 3 in the t^{th} iteration.

Since that $\tilde{\ell}(\boldsymbol{\theta}^{(t)})$ is monotonously increasing as t increases and $\tilde{\ell}(\boldsymbol{\theta}^{(t)})$ is bounded by the maximized log likelihood $\ell(\theta^{**}) = \sup_{\boldsymbol{\theta}} \ell(\boldsymbol{\theta})$, according to the monotone convergence theorem, $\{\tilde{\ell}(\boldsymbol{\theta}^{(t)})\}$ would converge as $t \to \infty$, implying that our algorithm would let $\{\boldsymbol{\theta}^{(t)}\}$ converges to a point $\boldsymbol{\theta}^*$ where $\tilde{\ell}(\boldsymbol{\theta})$ achieve a local mode. We are also aware that $\ell(\boldsymbol{\theta})$ may not achieve a local mode at $\boldsymbol{\theta}^*$.

2.6 Experiments

We illustrate our model based on both synthetic data and cervical screening test data.

2.6.1 Synthetic Data

In this section, we generate data from a simple hierarchical inhomogeneous hidden Markov model on the time interval $(0, 10)$. It contains two different transition structures A and B on two states. The transition structure of A and B is summarized by the transition rates, q_i, in Table 2.1.

The emission mechanism is modeled by simple categorical distributions: $p(O|S_1) \sim$

Table 2.1: Synthetic setting for transition parameters.

Structure	$q_1(t < 5)$	$q_2(t < 5)$	$q_1(t > 5)$	$q_2(t > 5)$
A	0.1	0.1	1	1
B	1	1	0.1	0.1

$\text{Ber}(p_1 = 0.95)$ and $p(O|S_2) \sim \text{Ber}(p_2 = 0.05)$.

We sample 200 time series from structure A and 300 time series from structure B. All time series are assumed to start at state 1 and have 50 observations randomly located on the time interval $(0, 10)$.

We assume a non-informative prior of model indicator $z_n \sim \text{Ber}(0.5)$ and set the maximum iteration number of EM, 50. We validate model results by computing the inference results of model frailty and hidden states in Table 2.2.

Table 2.2: Inference results of model model frailty and hidden states. Inference metrics for hidden states are summarized with mean and standard deviation among all time series.

	Predictive Accuracy	Precision	Recall
Model Frailty	0.946	0.9105	0.960
Hidden States	0.979(0.027)	0.940(0.168)	0.937(0.167)

To acquire the estimates and uncertainty of model parameters, we compute the mean and standard deviation of model parameters with 10 bootstrap resamples. The transition parameter estimates are summarized in Table 2.3 and emission parameter estimates p_1 and p_2 are summarized by mean and standard deviation 0.954(0.003) and 0.040(0.004), respectively.

Table 2.3: Bootstrapping results of transition parameters.

Structure	$q_1(t < 5)$	$q_2(t < 5)$	$q_1(t > 5)$	$q_2(t > 5)$
A	0.06(0.01)	0.05(0.02)	1.14(0.11)	1.11(0.11)
B	0.89(0.11)	1.01(0.11)	0.12(0.01)	0.07(0.02)

For comparison, we use Recurrent neural network models (RNN), which have been found to perform well with variable-length time series and capture the temporal correlation well because of their flexibility and lack of the Markov property (Choi et al., 2016). Variants of RNNs have been proposed to better balance memory needs and new features, including the Long Short-Term Memory (LSTM) architecture (Cho et al., 2014) and the gated recurrent neural network (GRU) (Chung et al., 2014). We compare RNN models with our proposed model on the same prediction task to illustrate that our model outperforms RNNs because of interpretable modeling of irregular samples and robustness to imbalanced data via latent clustering (hierarchical structure). We carry out a prediction task on the observation on the last time given all observations before. We generate 1000 testing time series with length 50, in which 500 of them are generated from structure A and the other 500 are generated from structure B. For the RNNs, we fit both a small model with layer size 16 and large model with layer size 64. The prediction results are shown in Table 2.4. In general, there exists some research on selecting the optimal number of neurons via Bayesian network, but it is out of the scope of this project.

Table 2.4: Model prediction for the observation of the last time in terms of Accuracy(ACC), Area Under The Curve(AUC), F1, Average Precision (AP), Precision (P), Recall(R).

Method	ACC	AUC	F1	AP	P	R
LSTM (small)	0.802	0.854	0.805	0.740	0.794	0.816
LSTM (large)	0.783	0.856	0.786	0.720	0.776	0.796
stacked LSTM (small)	0.820	0.886	0.819	0.765	0.826	0.812
stacked LSTM (large)	0.766	0.858	0.767	0.704	0.765	0.768
GRU (small)	0.814	0.865	0.812	0.759	0.822	0.802
GRU (large)	0.802	0.867	0.803	0.742	0.801	0.804
HIHMM	**0.859**	**0.910**	**0.858**	**0.809**	**0.862**	**0.853**

2.6.2 Screening Data

We demonstrate the HIHMM on our motivating cervical cancer screening test dataset from the Cancer Registry of Norway. Data used in the analyses will be available on request from the Cancer Registry of Norway, given legal basis according to the GDPR.

Data and Model Explanation

This dataset contains 1.7 million patients' screening testing records. Each patient has a censored observation at the last time stamp t_c, denoted by O_c, which indicates whether the woman is dead or alive at time t_c. Each patient has treatment indices to show when and how many treatments occurred, and results of screening tests for each of cytology, histology and HPV. Cytology and histology have four levels of outcomes: no risk, low risk, high risk and cancer, while HPV has two levels: negative and positive. Individual records are irregularly sampled with time recorded monthly for confidentiality, which implies an observation may have multiple results for one test at one timestamp.

We set four types of states: Normal, Low grade, High grade and Cancer and set $Z = 2$. Model transition structures are displayed in Figure 2.2. Model $\mathcal{M}_0$ refers to low-risk patients while Model $\mathcal{M}_1$ refers to high-risk patients. In $\mathcal{M}_0$, patient's state only transits between Normal and Low grade while the latent state in $\mathcal{M}_1$ is available to transition to High grade or Cancer. Due to the mechanism of disease progression, any state may only transit to its consecutive state and the Cancer state cannot transit back to High grade. All states are available to transit to Death.

Since the cancer progression strongly depends on age, for model z, the initial state is modeled as $S_{z1}|a_1 \sim \text{Cat}(\boldsymbol{\pi}_z(\mathcal{A}, a_1))$ and $\boldsymbol{\pi}_{zi} \sim \text{Dir}(\boldsymbol{\alpha}_{zi})$, where a_1 denotes

the age at the first screening test and $\mathcal{A}$ is a disjoint partition of observable ages, $\boldsymbol{\pi}_z(\mathcal{A}, a) = \boldsymbol{\pi}_{zi}$ if and only if $a \in \mathcal{A}_i$, and $\boldsymbol{\alpha}_{zi} \in \mathbb{R}^{+M_z}$ for i = 1, . . . , I.

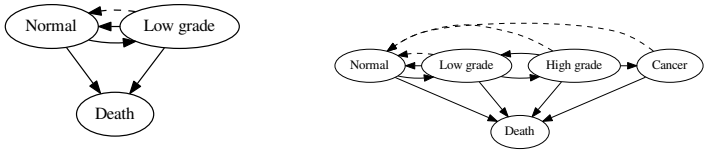

Figure 2.2: Transition structure of model $\mathcal{M}_0$ and $\mathcal{M}_1$. Solid lines denote the intensity transition while dashed lines denote that any state comes back to the normal state once treatment is completed.

The observations $\boldsymbol{O}$ have two levels: the number of screening tests $\boldsymbol{E}$ and the results of screening tests $\boldsymbol{G}$. Omitting the subscripts n and t, given state s, observations are modeled as

$$\begin{aligned} E_k|s &\sim \text{Poisson}(\eta_{sk})\,, \\ \boldsymbol{G}_k|E_k, s &\sim \text{Multinomial}(E_k, \tilde{\boldsymbol{\pi}}_{sk})\,, \\ \tilde{\boldsymbol{\pi}}_{sk} &\sim \text{Dir}(\tilde{\boldsymbol{\alpha}}_{sk})\,, \end{aligned}$$

where $\tilde{\boldsymbol{\alpha}}_{sk} \in \mathbb{R}^{+L_k}$ are hyper-parameters for observation model. The censored observations (dead/alive) are modeled by

$$p(O_c|S_T) = \begin{cases} P(t_T, t_c)_{S_T,\text{death}} & \text{if } O_c = \text{death}\,, \\ 1 - P(t_T, t_c)_{S_T,\text{death}} & \text{if } O_c \neq \text{death}\,. \end{cases}$$

The age partition is usually chosen in one of two ways. One is to divide age by quantiles of data to guarantee each segment has enough data for training. The other approach is to set the knots through expert knowledge. We decide the segmentation based on HPV information and choose the age partition as $\mathcal{A}$ as

$[0, 23)$, $[23, 30)$, $[30, 60)$ and $[60, \infty)$.

Specifically, since HPV status is one of most important indicators for cervical censor, we segment the age interval based on the empirical density of ages at which patients are found positive for HPV. The empirical density is estimated based on 100000 patients randomly sampled from the pool using gaussian kernel estimation. Then we fit the density of ages using a discontinuous piece-wise linear function with different numbers of intervals.

Fitting information is summarized in Table 2.5. We plot optimal sum of square errors under different N in Figure 2.3. From the figure, it visually shows that $N = 4$ is the optimal number of segmentation based on elbow criteria. Then combining the expert's opinion, we set the corresponding cutting points as 23, 30 and 60.

Table 2.5: Discontinuous piece-wise linear fitting under different numbers of intervals N. Optimal sum of square errors (SSE) and cutting points (CPs) are given.

N	SSE	CPs
2	3.88e-3	24.8
3	1.78e-3	24.7, 54.7
4	1.23e-3	25.2, 35.6, 60.6
5	0.93e-3	24.5, 26.2, 30.7, 67.2
6	0.78e-3	24.1, 25.2, 32.8, 56.9, 64.4

In the proposed learning approach, we set the number of EM iterations at $N_{\text{EM}} = 100$, and in the Limited-memory BFGS (L-BFGS) approach we set the number of optimization iterations as $N_{\text{L-BFGS}} = 8$. Automatic differentiation is implemented using the autograd package (Maclaurin, 2016) in Python.

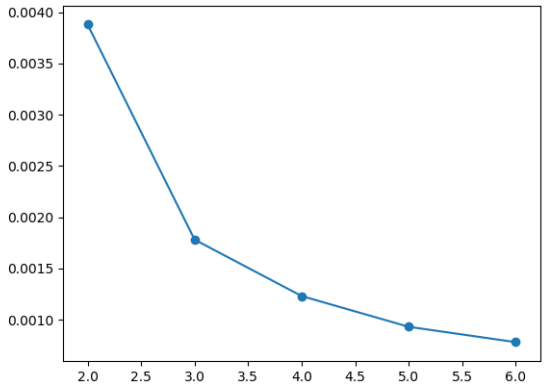

Figure 2.3: Sum of square errors under different number of segmentation N.

Table 2.6: Model prediction for the status of the last visit in terms of Accuracy (ACC), Area Under The Curve(AUC), F1, Average Precision (AP), Precision (P), Recall (R).

Method	ACC	AUC	F1	AP	P	R	training time (h)
LSTM (small)	0.9905	0.4939	0.0000	0.0095	0.0000	0.0000	1.11
LSTM (large)	0.9925	0.8563	0.4275	0.2359	0.7778	0.2947	3.38
stacked LSTM (small)	0.9914	0.8561	0.2773	0.1273	0.6875	0.1737	2.26
stacked LSTM (large)	**0.9926**	0.8573	0.4335	0.2409	**0.7808**	0.3000	7.5
GRU (small)	0.9920	0.8379	0.4089	0.2083	0.6962	0.2895	0.79
GRU (large)	0.9921	0.8678	0.4207	0.2178	0.7037	0.3000	2.30
CTIHMM	0.9910	0.9128	0.3466	0.1465	0.5517	0.2526	3.61
HIHMM	0.9914	0.9190	**0.5210**	**0.2774**	0.5589	**0.4895**	6.97
HIHMM (fast)	0.9912	**0.9268**	0.5014	0.2583	0.5466	0.4632	0.90

Model Comparison

We randomly select 80,000 patients' records for training and select another 20,000 records for testing. We compare different models by evaluating prediction on the status at the last visit. If a patient has at least one result whose level is greater than 1, then the status is defined as high risk denoted as 1. Otherwise, the status is defined as low risk denoted as 0, making it a binary classification problem.

The prediction procedure of our model is derived by Bayes rules. After model training, let the model parameter estimates be $\hat{\boldsymbol{\psi}}$. Given new patient histori-

cal records $\boldsymbol{O}^*$, compute the predictive distribution of model index $p(z^*|\boldsymbol{O}^*, \hat{\boldsymbol{\psi}})$. Next, given model index z, the predictive distribution of the state at the second to last visit is $p(S^*_{T-1}|z, \boldsymbol{O}^*, \hat{\boldsymbol{\psi}})$ derived from FFBS. Then the predictive distribution of the state of the last visit is $p(S^*_T|\boldsymbol{O}^*, \hat{\boldsymbol{\psi}}) = \sum_{s,z} p(z^* = z|\boldsymbol{O}^*, \hat{\boldsymbol{\psi}}) p(S^*_T|S^*_{T-1} = s, z, \hat{\boldsymbol{\psi}}) p(S^*_{T-1} = s|z, \boldsymbol{O}^*, \hat{\boldsymbol{\psi}})$ and given the number of screening tests at the last visit $\boldsymbol{E}^*_T$, the predictive distribution of screening test results is $p(\boldsymbol{G}^*_T|\boldsymbol{O}^*, \boldsymbol{E}^*_T, \hat{\boldsymbol{\psi}}) = \sum_s p(S^*_T = s|\boldsymbol{O}^*, \hat{\boldsymbol{\psi}}) p(\boldsymbol{G}^*_T|S^*_T = s, \boldsymbol{E}^*_T, \hat{\boldsymbol{\psi}})$. Finally, the predictive distribution of the last status is $G^* \sim \text{Ber}(p^*)$, where $p^* = p\left(\sum_{i=0}^{1} \sum_{j=2}^{3} \boldsymbol{G}^*_T[i,j] \geq = 1|\boldsymbol{O}^*, \boldsymbol{E}^*_T, \hat{\boldsymbol{\psi}}\right)$, and it is estimated by $\hat{G}^* = 1$ if $p^* \geq 0.5$ and 0 otherwise. Similar procedures are available for the CTIHMM.

We also compare our model to RNNs, where each patient's record is modeled as one time-series and the features at each visit include patient age, screening result and treatment indicator. The screening result of patient n at the t^{th} visit is $\vec{\boldsymbol{G}}_{n,t}$. The treatment indicator is equal to 1 if and only if the patient has accepted treatment. We again compare to LSTM, stacked LSTM, and GRU with the same small and large layer sizes as in the synthetic experiment. For stacked LSTM, two LSTMs are stacked. For the HIHMM, through cross validation, the model prior is set as $p = 0.001$ and we set 100 iterations in the EM algorithm. In addition, we try a fast variant of EM using only 10 iterations. For the CTIHMM, we also set 100 iterations in the EM algorithm. We summarize both prediction results and training time in Table 2.6. It shows our model outperforms state of the art methods overall on the set of criteria: Area Under the Curve (AUC), F1 value (F1), Average Precision (AP) and Recall (R). Those metrics related to recall score are important in clinic diagnosis, it is better to mis-classify low-risk patients rather than high-risk patients. We also show our model has competitive running time compared with neural network models.

2.6.3 Model Validation

Based on epidemiological studies, the incidence rate of cervical cancer can provide us with guidance on how to choose priors for the frailty rate of a given population (Bray et al., 2018). An informative prior is indeed preferable here, so we set a conservative model index prior $p = 0.2$. We note that there is always a trade off between precision and recall, and $p = 0.2$ provides a reasonable balance based on model comparison results. We present two types of results on population-level data. First we present the MLEs for all model parameters along with bootstrapped standard deviations. Second we perform model validation using Kaplan-Meier estimators as suggested in Titman and Sharples (2008) and predictive accuracy via proposed average posterior predictive probability.

We randomly divide all data into clusters such that each cluster has 100 individual observation sequences. Using a bootstrap technique, we randomly select 2400 clusters with replacement for model inference. We independently repeat the same inference on different selections 5 times. Inference results are discussed as follows:

After EM converges, according to (2.35) in the paper, we summarized the quantiles of estimated posterior probability of Model 1 in Table 2.7. It shows that more than 75% of women have a posterior probability of belonging to the high-risk model $\mathcal{M}_1$ that is less than the prior probability $p = 0.2$. This suggests that more than 75% are likely to be in the low-risk disease exposure category according to their screening test results. Moreover, it also justifies the expert knowledge that around 20% belong to the high-risk disease exposure category. Whereas, the 90% credible interval of the posterior hyper-parameter $\tilde{p}$ is $(0.1825, 0.2358)$, which is still close to the prior hyper-parameter $p = 0.2$. This implies that the observations do not affect the posterior of the model indexes significantly and

choosing a reasonable prior is important. This is likely an artifact of the dataset, which is highly skewed towards normal test results. More balanced datasets may exhibit less sensitivity to prior specification.

Estimates related to the emission mechanism are summarized in Table 2.8 and Table 2.9. Table 2.8 shows the estimates of diagnostic test result probabilities conditional on hidden state. The estimates of diagnostic test results match the definition of states. The more advanced a patient's disease state, the more likely she is to get an abnormal screening result. And the small standard deviations suggest that our data are sufficient to get precise estimates of the emission parameters. Table 2.9 shows the number of screening tests for women in different states. Due to the fact that the expectation of a Poisson distribution is exactly the Poisson intensity parameter, the Poisson intensities show that individuals at normal state are more likely to be assigned to a cytology screening test. Individuals in abnormal disease states (low-grade, high-grade and cancer) are more likely to be given histology and HPV tests. This result matches what is expected in clinical practice in that women will be assigned more precise screening tests as they present more severe symptoms.

The initial state's information is summarized in Table 2.10 which shows the initial information of the population categorized by the specified age partition for model $\mathcal{M}_0$ and model $\mathcal{M}_1$. From a model specification perspective, women in the low-risk model, $\mathcal{M}_0$, are assumed to be more likely to stay at a normal state than women in the high-risk model $\mathcal{M}_1$, regardless of their ages. On the other hand, in the high-risk model, $\mathcal{M}_1$, women in the age interval $(23, 30)$ are most likely to belong to a high risk state at the initial screening test.

Finally, Table 2.11 displays the estimates of transition intensities in the two models. It shows patients are more likely to transition from the low grade state

to the normal state, whether or not they are in model $\mathcal{M}_0$ or model $\mathcal{M}_1$. On the other hand, λ_{34} has significantly higher standard deviations than other intensity parameters because of the scarcity of data for individuals with cancer who died during the period in which the data was collected.

Table 2.7: Quantiles of maximum likelihood estimates of posterior probability of Model 1.

quantiles	0.05	0.25	0.5	0.75	0.95
$\tilde{p}$	0.1825(0.0004)	0.1936(0.0001)	0.1966(0.0002)	0.1988(0.0001)	0.2358(0.0049)

Table 2.8: Maximum likelihood estimates of diagnostic test result probabilities conditioned on hidden state.

cytology				
state	0	1	2	3
normal	1.0000(0.0000)	0.0000(0.0000)	0.0000(0.0000)	0.0000(0.0000)
low grade	0.0331(0.0011)	0.8021(0.0020)	0.1631(0.0019)	0.0016(0.0001)
high grade	0.0614(0.0068)	0.0054(0.0010)	0.9198(0.0059)	0.0133(0.0012)
cancer	0.0619(0.0125)	0.0567(0.0084)	0.6254(0.0158)	0.2560(0.0097)

histology				
state	0	1	2	3
normal	0.9879(0.0004)	0.0108(0.0004)	0.0007(0.0001)	0.0006(0.0001)
low grade	0.2621(0.0023)	0.1561(0.0006)	0.5810(0.0030)	00009(0.0002)
high grade	0.0345(0.0018)	0.0068(0.0015)	0.9573(0.0026)	0.0014(0.0001)
cancer	0.0263(0.0062)	0.0133(0.0015)	0.0796(0.0014)	0.8808(0.0066)

HPV		
state	-	+
normal	0.9966(0.0010)	0.0034(0.0010)
low grade	0.3720(0.0048)	0.6280(0.0048)
high grade	0.0357(0.0054)	0.9643(0.0054)
cancer	0.0194(0.0019)	0.9806(0.0019)

For model validation we randomly select 2400 clusters of data in which each cluster has 100 individual sequences of observations. We implement both the HIHMM and the CTIHMM for the same dataset. We follow the method proposed in Titman and Sharples (2008) that utilizes Kaplan-Meier estimators to validate

Table 2.9: Maximum likelihood estimates of Poisson intensities for the number of tests conditioned on true state.

state	cytology	histology	HPV
normal	0.9880(0.0002)	0.0140(0.0001)	0.0053(0.0001)
low grade	0.7912(0.0008)	0.1856(0.0018)	0.1003(0.0016)
high grade	0.4595(0.0024)	0.6493(0.0018)	0.0290(0.0017)
cancer	0.5091(0.0191)	0.8627(0.0322)	0.0278(0.0023)

Table 2.10: Maximum likelihood estimates of the probability of being a particular state at the time of the first screening.

age range	16-23	23-30	30-60	60-
Model 0				
normal	0.9315(0.0010)	0.9415(0.0017)	0.9614(0.0007)	0.9643(0.0009)
low grade	0.0685(0.0010)	0.0585(0.0017)	0.0386(0.0007)	0.0357(0.0009)
Model 1				
normal	0.9187(0.0015)	0.9040(0.0011)	0.9287(0.0006)	0.9308(0.0020)
low grade	0.0761(0.0013)	0.0677(0.0022)	0.0438(0.0011)	0.0354(0.0016)
high grade	0.0041(0.0003)	0.0271(0.015)	0.0262(0.0007)	0.0263(0.0018)
cancer	0.0011(0.0001)	0.0013(0.0002)	0.0013(0.0002)	0.0075(0.0006)

continuous-time HMMs. We define failure as the first observation of a high-risk or cancer test result directly following an initial normal or low-grade test result. Accurately predicting this time-to-event is of practical importance because clinical intervention is only possible in the high-grade state. Treating patients at this stage is critical to preventing precancerous lesions from progressing to cervical cancer.

The empirical Kaplan-Meier estimator is an important criterion because it measures prediction on the whole process rather than only the last visit, and it is defined as $\hat{S}(t) = \prod_{i:t_i \leq t} (1 - d_i/n_i)$, where t_i is a time when at least one failure is observed, d_i is the number of failures that occurred at time t_i, and n_i is the number of individuals known to have survived up to time t_i. We randomly choose 24,000 records to generate an empirical Kaplan-Meier estimator and to simulate 100 sequences from both the CTIHMM and HIHMM with 100 times repetitions

Table 2.11: Maximum likelihood estimates for age dependent transition intensities.

age range	16-23	23-30	30-60	60-
Model 0				
λ_{01}	0.1718(0.0061)	0.0809(0.0017)	0.0546(0.0006)	0.0439(0.0015)
λ_{02}	0.0005(0.0000)	0.0018(0.0000)	0.0019(0.0000)	0.0147(0.0001)
λ_{10}	1.7064(0.0327)	1.2637(0.0292)	0.4893(0.0118)	2.2169(0.0641)
λ_{12}	0.0024(0.0002)	0.0021(0.0001)	0.0011(0.0001)	0.0122(0.0013)
Model 1				
λ_{01}	0.1938(0.0065)	0.1191(0.0032)	0.0730(0.0011)	0.0536(0.0020)
λ_{04}	0.0014(0.0001)	0.0015(0.0001)	0.0015(0.0001)	0.0121(0.0002)
λ_{10}	1.6854(0.0295)	1.3541(0.0358)	1.6331(0.0131)	2.3063(0.1288)
λ_{12}	0.0815(0.0086)	0.2276(0.0015)	0.1867(0.0030)	0.2395(0.0090)
λ_{14}	0.0058(0.0003)	0.0048(0.0002)	0.0032(0.0003)	0.0150(0.0006)
λ_{21}	0.3585(0.0593)	0.0780(0.0067)	0.0663(0.0052)	0.2245(0.0288)
λ_{23}	0.0720(0.0095)	0.0307(0.0017)	0.1012(0.0060)	0.5166(0.0366)
λ_{24}	0.0150(0.0004)	0.0069(0.0007)	0.0034(0.0003)	0.0166(0.0006)
λ_{34}	1.0366(0.0611)	2.5642(0.5100)	2.6911(0.1251)	1.6805(0.3432)

for credible interval. Simulation details are proposed as follows:

To simulate one Kaplan-Meier curve from a HIHMM, we propose the following procedures:

1 First, reduce all individuals' ages at their first screening test to a set A_1, and categorize individuals' time intervals between two consecutive screening tests into four sets denoted as $\tilde{I}_i$ for $i = 1, \ldots 4$. Label the four screening testing results, Nornal, Low grade, High grade and Cancer as $1, 2, 3, 4$ sequentially as their score. Then any time interval (a, b) is categorized into $\tilde{I}_i$, if and only if the largest score of both cytology and histology screening test results at time a is i. For each set $\tilde{I}_i$, we map elements of $\tilde{I}_i$ to their corresponding lengths and name the new set as I_i. Also, we reduce all posterior probabilities of model indexes into a set $\tilde{P}$.

2 Second, we randomly sample an initial age a_1 from A_1 and randomly sample

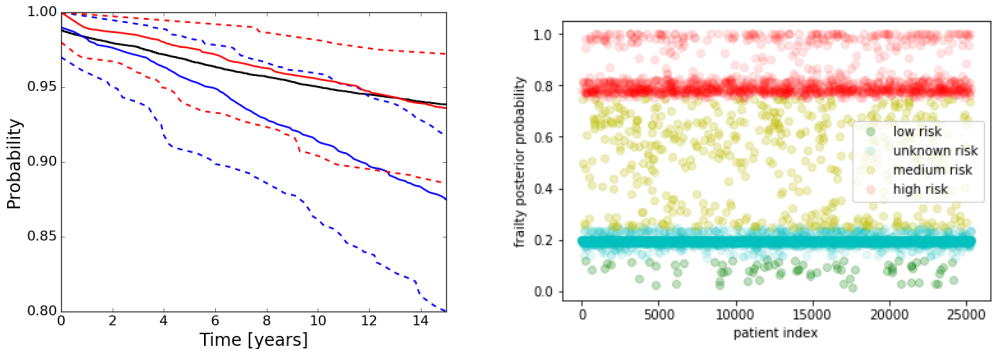

Figure 2.4: Top panel: Empirical Kaplan-Meier curve (black) and simulated Kaplan-Meier curves, which are summarized using the 95% credible interval (dashed lines) and the median (solid lines), from the CTIHMM (blue) and HIHMM (red). Bottom panel: Posterior probabilities of belonging to the frailty class for each individual from a test set. Risk stratification is possible by thresholding the probabilities. Threshold probabilities in this example are $(0, 0.125, 0.25, 0.75, 1)$. Color indicates falling between two probability thresholds.

a posterior probability of model index $\tilde{p}$ from $\tilde{P}$.

3 Then sample model index z via $z \sim \text{Ber}(\tilde{p})$ and sample an initial state $S_1 \sim \text{Cat}(\hat{\boldsymbol{\pi}}_z(\mathcal{A}, a_1))$, where $\hat{\boldsymbol{\pi}}_{zi} = \hat{E}(\boldsymbol{\pi}_{zi}) = \frac{\hat{\boldsymbol{\alpha}}_{zi}}{\sum \hat{\boldsymbol{\alpha}}_{zi}}$.

4 Sequentially sample states until the State S_T fails according to the failure definition of Kaplan-Meier estimator. Specifically, based on current state S_{t-1} and current age a_{t-1} , sample the screening time interval Δ_{t-1} from $I_{S_{t-1}}$ and denote $a_t = a_{t-1} + \Delta_{t-1}$. Then compute transition matrix $P([a_{t-1}, a_t]|\mathcal{A}, \hat{\boldsymbol{\lambda}}_z)$ from age a_{t-1} to a_t. Finally sample the current state S_t via $S_t \sim \text{Cat}(P[a_{t-1}, a_t]_{S_{t-1},:})$.

5 Compute the failure time by $F = \sum_{t=1}^{T} \Delta_t$.

6 Repeat [2] to [5] M times. We obtain M failure times and then order them as a sequence $\{F_m\}_{m=1}^{M}$.

7 Based on the simulated failure times $\{F_m\}_{m=1}^{M}$, the simulated Kaplan-Meier

curve is obtained through $S(t) = 1 - \frac{1}{M}\sum_{t \geq F_m} 1$.

Multiple Kaplan-Meier curves are simulated by independently repeating the above procedures.

Figure 2.4 shows the empirical Kaplan-Meier curve in black, simulated Kaplan-Meier curves from the CTIHMM in blue, and simulated Kaplan-Meier curves from the HIHMM in red. Solid lines denote the median simulated curve and dashed lines denote the 95% credible intervals based on the 100 replications. The results show that the empirical Kaplan-Meier curve is always near the median and within the 95% credible intervals generated by the HIHMM. This is not the case with the CTIHMM. In this sense the HIHMM outperforms the CTIHMM in an important clinical metric.

The HIHMM has a relatively high Kaplan-Meier estimate at time 0 because the informative prior $p = 0.2$ is relatively small, which makes simulated patients more likely be in the low-risk model $\mathcal{M}_\prime$ at the initial time. Moreover, these patients are more likely to stay at the normal state for longer. However, the trend of the median curve from the HIHMM more closely tracks that of the empirical Kaplan-Meier curve, compared with the trend of the median curve from the CTIHMM. This suggests that the HIHMM models disease progression better than the CTIHMM. The Kaplan-Meier curves simulated from the CTIHMM are always underestimated.

We also propose average posterior predictive probability for Cytology, Histology and HPV at the last visit given the screening tests as a quantitative measurement for model validation. For each patient, the posterior predictive probabilities are $p(O^*_{T,\text{test}}|O^*_{-T}, \hat{\boldsymbol{\psi}}, E^*_T)$ where tests include Cytology, Histology and HPV. We take 240000 patients for training and 20000 patients for testing and model validation results for CTIHMM and HIHMM are summarized in Table 2.12. The results

illustrate that HIHMM outperforms CTIHMM in model prediction, especially for HPV which has less records for training.

Table 2.12: Average posterior predictive probabilities of Cytology, Histology and HPV for CTIHMM and HIHMM models

Model	Cytology	Histology	HPV
CTIHMM	0.9563	0.7597	0.6550
HIHMM	**0.9571**	**0.7613**	**0.7010**

2.7 Conclusion and Discussion

One application of the HIHMM in the context of population-based screening programs is risk stratification of the population. The latent random variable z_n is an indicator of belonging to a frailty class in the population. Given the fitted model parameters $\boldsymbol{\psi}$ it is possible to compute the posterior probability of belonging to the frailty class for individual women, i.e., given an observed sequence of test results $\boldsymbol{O}_n$ and model parameters $\boldsymbol{\psi}$, the posterior predictive distribution $p(z_n|\boldsymbol{\psi}, \boldsymbol{O}_n)$ gives a measure of the likelihood of an individual to be at risk of developing cervical cancer conditioned on their observed test results. Such information could be used to more efficiently screen a population by avoiding the over screening of women at low-risk and the under screening of women at high-risk.

Examples of these posterior probabilities are shown in Fig 2.4. For illustration purposes, we have chosen risk thresholds of $\{0.125, 0.25, 0.75\}$ with the following interpretation.

$$
\begin{aligned}
0 \leq p(z_n|\boldsymbol{\psi}, \boldsymbol{O}_n) < 0.125 &\implies \text{low-risk} \\
0.125 \leq p(z_n|\boldsymbol{\psi}, \boldsymbol{O}_n) < 0.25 &\implies \text{unknown risk} \\
0.25 \leq p(z_n|\boldsymbol{\psi}, \boldsymbol{O}_n) < 0.75 &\implies \text{medium-risk} \\
0.75 < p(z_n|\boldsymbol{\psi}, \boldsymbol{O}_n) \leq 1 &\implies \text{high-risk}
\end{aligned}
$$

Two main clusters are apparent in the data corresponding to unknown risk and high risk. The unknown risk cluster is those patients close to the prior probability of 20%. These patients lack sufficient observations to make an informed decision about their risk profile. This suggests these patients should be followed up with the standard screening protocol. The high risk cluster is those patients who are more likely to be in a high-grade state. This suggests these patients may require immediate follow up. The two smaller clusters of low risk and medium risk are comprised of patients that may require decreased or increased screening frequencies, respectively, relative to the standard screening protocol.

In summary, this chapter has made the following contributions:

- We model treatment effects in CTIHMM and make CTIHMM inference possible for population-level datasets by using piece-wise constant intensity functions and deriving a scalable EM-based inference algorithm.
- We put a hierarchical structure over CTIHMM to explain population heterogeneity in terms of frailty but share the same states and their emission probability, which makes the model more practical, resulting in our HIHMM.
- We utilize prior distributions in the model to achieve more accurate estimates when dealing with imbalanced data.
- We perform full model inference on a cancer screening dataset and show that modeling population heterogeneity improves performance in terms of

Kaplan-Meier estimators and proposed average posterior predictive probability.

- We illustrate how the model may be used to better inform public health professionals by providing a risk stratification mechanism.

Chapter 3

Regularization of Sparse Gaussian Processes

In this chapter, we propose a regularization framework for inducing-point based sparse Gaussian process models (SGP) and extend this framework into latent variable models, balancing the distribution of inducing inputs and embedding inputs, and leading to better model prediction. We theoretically justify the use of this regularization framework by proving that performing variational inference (VI) with our regularization term is equivalent to directly performing VI on a related empirical Bayes model with a prior on its inducing inputs.

The rest of this chapter is organized as follows. We first review existing literature of Gaussian process and sparse Gaussian process in Section 3.1. In Section 3.2, we show the motivation of our regularization framework in SGP. To take a better Gaussian process approximation and a better model fitting, it is necessary to consider both marginal likelihood and the distribution of sampling inputs. We propose two regularizers from non-parametric and parametric aspects respectively. Then, we extend our regularization framework to latent variable models and justify it through a related hierarchical empirical Bayesian model in

Section 3.3. In Section 3.4, we illustrate the importance of our regularization framework using three different real datasets. Finally, Section 3.5 summarizes our work and discusses its benefits.

3.1 Preliminaries

A Gaussian process is a distribution over functions f satisfying the assumption that given any collection of inputs $\{x_n\}_{n=1}^N$, the distribution of the corresponding output $[f(x_1), f(x_2), \ldots, f(x_n)]$ is a multivariate Gaussian distribution. Gaussian processes are widely used to model any smooth functions because of the convenience of the Gaussian distribution.

In Section 3.1.1, we explore Gaussian process from both frequentist aspect and Bayesian aspect. We discuss different types of covariance functions, nugget effects and latent inputs. Section 3.1.2 discusses different types of sparse Gaussian process models which are directly based on low rank approximation. Its hyperparameter optimization is studied in Section 3.1.3. Full Bayesian inference and variational inference for sparse Gaussian processes are discussed in Section 3.1.4 and Section 3.1.5 separately. We also carry out multiple experiments for model comparison in Section 3.1.6.

3.1.1 Gaussian Process

This section introduces the definition and properties of Gaussian processes. First, Gaussian processes are introduced from a frequentist perspective and their properties are studied in Section 3.1.1. Then we discuss Gaussian processes from a Bayesian perspective in Section 3.1.1. Different covariance functions are discussed in Section 3.1.1. Section 3.1.1 discusses how the GP is used for dimensionality

reduction via latent variable models.

Gaussian Processes from a Generalization Aspect

Gaussian processes are an extension of the multivariate Gaussian distribution. It extends the multivariate Gaussian distribution from finite dimensions to infinite dimensions. For example, we consider a set of random variables $\boldsymbol{f} = \{f_i\}_{i\in\mathcal{X}} \sim \mathcal{N}(\boldsymbol{\mu}, K)$, where $\mathcal{X}$ is a sorted index set. The inference for multivariate Gaussian distribution is tractable because of two properties of the multivariate Gaussian distribution.

The first property is that any marginal distribution is a multivariate Gaussian distribution. Mathematically, considering any two disjoint index sets A and B, assume the corresponding random variables are $\boldsymbol{f}_A$ and $\boldsymbol{f}_B$ respectively. For any index set $\boldsymbol{S}$, we denote $\boldsymbol{\mu}_{\boldsymbol{S}}$ and $K_{\boldsymbol{S},\boldsymbol{S}}$ as the mean and covariance matrix of $\boldsymbol{f}_{\boldsymbol{S}}$. For any two index sets $\boldsymbol{S}$ and $\boldsymbol{S}^*$, $K_{\boldsymbol{S},\boldsymbol{S}^*}$ denotes the covariance matrix between $\boldsymbol{S}$ and $\boldsymbol{S}^*$. Then the marginal distributions $p(\boldsymbol{f}_A)$ and $p(\boldsymbol{f}_B)$ have expressions

$$p(\boldsymbol{f}_A) = \int p(\boldsymbol{f}_A, \boldsymbol{f}_B) d\boldsymbol{f}_B = \mathcal{N}(\boldsymbol{\mu}_A, K_{AA}), \tag{3.1}$$

$$p(\boldsymbol{f}_B) = \int p(\boldsymbol{f}_A, \boldsymbol{f}_B) d\boldsymbol{f}_A = \mathcal{N}(\boldsymbol{\mu}_B, K_{BB}). \tag{3.2}$$

The second property is that any conditional distribution is a multivariate Gaussian distribution. Specifically, the conditional distributions are derived as

$$p(\boldsymbol{f}_A|\boldsymbol{f}_B) = \mathcal{N}(\boldsymbol{\mu}_A + K_{AB}K_{BB}^{-1}(\boldsymbol{f}_B - \boldsymbol{\mu}_B), K_{AA} - K_{AB}K_{BB}^{-1}K_{BA}), \tag{3.3}$$

$$p(\boldsymbol{f}_B|\boldsymbol{f}_A) = \mathcal{N}(\boldsymbol{\mu}_B + K_{BA}K_{AA}^{-1}(\boldsymbol{f}_A - \boldsymbol{\mu}_A), K_{BB} - K_{BA}K_{AA}^{-1}K_{AB}). \tag{3.4}$$

Due to these two properties, any inference for multivariate Gaussian distributions becomes tractable. Gaussian processes are an extension of the multivariate

Gaussian distribution. They extend finite dimensions of $\boldsymbol{f}$ to infinite dimensions, at the same time they also guarantee those two important properties.

A multivariate Gaussian distribution is fully specified by mean $\boldsymbol{\mu}$ and variance-covariance matrix K. A Gaussian process is fully specified by mean function $\mu(x)$ and covariance function $k_f(x, x')$. Therefore, given any set of inputs, the output can be fully specified as a multivariate Gaussian distribution.

Definition 3.1.1. A Gaussian process is a collection of random variables, any finite number of which have a joint multivariate Gaussian distribution. It is written as

$$f \sim \mathcal{GP}(\mu(x), k_f(x, x')) \tag{3.5}$$

where x and x' are inputs from the input space.

Generally, the mean function is set to 0 or is modeled as a constant function or a linear function and the variance-covariance function is modeled via a squared exponential covariance function, Matérn covariance and so on, according to the characteristics of the specific dataset. In this dissertation, we assume that $\mu(x) \equiv 0$ unless otherwise stated.

For finite inputs $\boldsymbol{x}$, the prior distribution depends on the variance-covariance matrix K which is derived from covariance function k_f. It is a multivariate Gaussian distribution. Letting $K_{\boldsymbol{x},\boldsymbol{x}} = k_f(\boldsymbol{x}, \boldsymbol{x})$ and $\boldsymbol{f} = f(\boldsymbol{x})$, the distribution is expressed as

$$p(\boldsymbol{f}|\boldsymbol{x}) = \mathcal{N}(\mathbf{0}, K_{\boldsymbol{x},\boldsymbol{x}})\,. \tag{3.6}$$

The covariance function k_f depends on kernel parameters $\boldsymbol{\theta}$ which are dropped for notation simplification.

In real applications, we assume observations are corrupted with white noise ϵ.

Then the observations are modeled by

$$\boldsymbol{y} = f(\boldsymbol{x}) + \boldsymbol{\epsilon}, \boldsymbol{\epsilon} \sim \mathcal{N}(\boldsymbol{0}, \sigma^2 \boldsymbol{I}), \tag{3.7}$$

After marginalizing the latent variable $\boldsymbol{f}$, the marginal distribution is expressed as

$$\begin{aligned} p(\boldsymbol{y}|\boldsymbol{x}) &= \int p(\boldsymbol{y}|\boldsymbol{f}) p(\boldsymbol{f}|\boldsymbol{x}) d\boldsymbol{f} \\ &= \mathcal{N}(\boldsymbol{y}|\boldsymbol{0}, K_{\boldsymbol{x},\boldsymbol{x}} + \sigma^2 \boldsymbol{I}). \end{aligned} \tag{3.8}$$

Then given observations $\boldsymbol{y}$ and corresponding inputs $\boldsymbol{x}$, the posterior Gaussian process is expressed as

$$f|\boldsymbol{y}, \boldsymbol{x} \sim \mathcal{GP}(\tilde{\mu}(\cdot), \tilde{k}_f(\cdot, \cdot)) \tag{3.9}$$

where given new inputs $\boldsymbol{x}^*$, the posterior mean vector is $\tilde{\mu}(\boldsymbol{x}^*) = K_{\boldsymbol{x}^*,\boldsymbol{x}}(K_{\boldsymbol{x},\boldsymbol{x}} + \sigma^2 \boldsymbol{I})^{-1} \boldsymbol{y}$ and the posterior variance-covariance matrix is $\tilde{k}_f(\boldsymbol{x}^*, \boldsymbol{x}^*) = K_{\boldsymbol{x}^*,\boldsymbol{x}^*} - K_{\boldsymbol{x}^*,\boldsymbol{x}}(K_{\boldsymbol{x},\boldsymbol{x}} + \sigma^2 \boldsymbol{I})^{-1} K_{\boldsymbol{x},\boldsymbol{x}^*}$.

The predictive distribution at new inputs $\boldsymbol{x}^*$ is derived as

$$\boldsymbol{f}^*|\boldsymbol{y}, \boldsymbol{x}, \boldsymbol{x}^* \sim \mathcal{N}(\tilde{\mu}(\boldsymbol{x}^*), \tilde{k}_f(\boldsymbol{x}^*, \boldsymbol{x}^*)), \tag{3.10}$$

$$\boldsymbol{y}^*|\boldsymbol{y}, \boldsymbol{x}, \boldsymbol{x}^* \sim \mathcal{N}(\tilde{\mu}(\boldsymbol{x}^*), \tilde{k}_f(\boldsymbol{x}^*, \boldsymbol{x}^*) + \sigma^2 \boldsymbol{I}). \tag{3.11}$$

Gaussian Processes from a Kernel Regression Aspect

Assume we have n observations $\boldsymbol{y}$, and each observation y_i matches corresponding covariates $\boldsymbol{x}_i$ with dimension s, then a kernel linear regression model

displays as

$$y_i = \phi(\boldsymbol{x}_i)^T \boldsymbol{w} + \epsilon_i \,, \forall i = 1, \ldots, n \,, \tag{3.12}$$

where $\phi(\boldsymbol{x})$ is a kernel function of covariates $\boldsymbol{x}$ and it is a mapping from $\mathbb{R}^s \to \mathbb{R}^m$, $\boldsymbol{w}$ is a weight vector with dimension m and $\epsilon_i \overset{iid}{\sim} \mathcal{N}(0, \sigma^2)$. From the Bayesian perspective, we consider a multivariate Gaussian prior for the weight vector $\boldsymbol{w} \sim \mathcal{N}(\boldsymbol{w}|\boldsymbol{0}, \Sigma_w)$. Then we marginalize the weight vector $\boldsymbol{w}$ to compute the likelihood function:

$$\begin{aligned} p(\boldsymbol{y}|\boldsymbol{x}) &= \int \mathcal{N}(\boldsymbol{y}|\phi^T(\boldsymbol{x})\boldsymbol{w}, \sigma^2 \boldsymbol{I}) \mathcal{N}(\boldsymbol{w}|\boldsymbol{0}, \Sigma_{\boldsymbol{w}}) d\boldsymbol{w} \\ &= \mathcal{N}(\boldsymbol{y}|\boldsymbol{0}, \sigma^2 \boldsymbol{I} + \phi^T(\boldsymbol{x}) \Sigma_{\boldsymbol{w}} \phi(\boldsymbol{x}))\, m. \end{aligned} \tag{3.13}$$

This is exactly analogous to the marginalization in Gaussian process (3.8). Kernel regression marginalizes the weight vector while Gaussian process marginalizes the latent variables $\boldsymbol{f}$. Because $\Sigma_{\boldsymbol{w}}$ is a covariance matrix which is positive definite, it can be decomposed as $\Sigma_{\boldsymbol{w}} = L_{\boldsymbol{w}}^T L_{\boldsymbol{w}}$ where $L_{\boldsymbol{w}}$ is a low triangular matrix. Then we rewrite $\phi^T(\boldsymbol{x})\Sigma\phi(\boldsymbol{x}) = \phi^T(\boldsymbol{x}) L_{\boldsymbol{w}}^T L_{\boldsymbol{w}} \phi(\boldsymbol{x}) = \tilde{\phi}^T(\boldsymbol{x})\tilde{\phi}(\boldsymbol{x})$, where $\tilde{\phi}(\boldsymbol{x}) = L_{\boldsymbol{w}} \phi(\boldsymbol{x})$.

Comparing (3.8) and (3.13), when $K_{x,x} = \tilde{\phi}^T(\boldsymbol{x})\tilde{\phi}(\boldsymbol{x})$, they are exactly equivalent. It illustrates that Kernel regression is a specific Gaussian process when the corresponding covariance function has a certain kernel function that $K_{x,x} = \tilde{\phi}^T(\boldsymbol{x})\tilde{\phi}(\boldsymbol{x})$.

Covariance Functions

Covariance function selection plays an crucial role in Gaussian processes because it encodes all assumptions about the function which we wish to learn. The

basic assumption about the covariance function is that for any set of inputs, their corresponding covariance matrix must be positive definite. Due to this assumption, the positive definite kernel function is defined. The formal definition is shown as follows.

Before giving the formal definition of covariance function, we introduce kernel function k. We need to note that this kernel is different from the kernel in kernel regression. It is a function mapping a pair of inputs $\boldsymbol{x} \in \mathcal{X}$ and $\boldsymbol{x}' \in \mathcal{X}$ to $\mathbb{R}$. Then we introduce the definition of a semi-positive definite kernel function.

Definition 3.1.2. A kernel function k is a positive semi-definite function if for any $n \in \mathbb{N}$, for all $\boldsymbol{x}_i \in \mathcal{X}$ and $a_i \in \mathbb{R}$, we have $\sum_{i=1}^{n} \sum_{j=1}^{n} a_i a_j k(\boldsymbol{x}_i, \boldsymbol{x}_j) \geq 0$.

This definition is derived by inducing a random variable $T = \sum_{i=1}^{n} a_i Z(\boldsymbol{x}_i)$ where Z is a random process with kernel k. Then variance of T is

$$\begin{aligned} \text{var}(T) &= \sum_{i=1}^{n} \sum_{j=1}^{n} a_i a_j \text{Cov}(Z(\boldsymbol{x}_i), Z(\boldsymbol{x}_j) \\ &= \sum_{i=1}^{n} \sum_{j=1}^{n} a_i a_j k(\boldsymbol{x}_i, \boldsymbol{x}_j)\,. \end{aligned} \tag{3.14}$$

To guarantee the positivity of variance of random variable T, it is necessary that kernel function must satisfy $\sum_{i=1}^{n} \sum_{j=1}^{n} a_i a_j k(\boldsymbol{x}_i, \boldsymbol{x}_j) > 0$ which means that kernel function must be semi-positive.

Then covariance function is defined as follow:

Definition 3.1.3. f is a covariance function if and only if f is a positive semidefinite kernel function.

Covariance Function properties

Covariance functions have some important properties:

- The sum of two covariance functions is a covariance function.

- The product of two covariance functions is a covariance function.
- Direct sum of two covariance functions is a covariance function. If $k(\boldsymbol{x}_1, \boldsymbol{x}_1')$ and $k(\boldsymbol{x}_2, \boldsymbol{x}_2')$ are covariance functions over $\mathcal{X}_1$ and $\mathcal{X}_2$ separately, $k(\boldsymbol{x}, \boldsymbol{x}') = k_1(\boldsymbol{x}_1, \boldsymbol{x}_1') + k_2(\boldsymbol{x}_2, \boldsymbol{x}_2')$ is a covariance function, where $\boldsymbol{x} = [\boldsymbol{x}_1, \boldsymbol{x}_2] \in \mathcal{X}_1 \times \mathcal{X}_2$ and $\boldsymbol{x}' = [\boldsymbol{x}_1', \boldsymbol{x}_2'] \in \mathcal{X}_1 \times \mathcal{X}_2$.
- Tensor product of two covariance functions is a covariance function. It means that $k(\boldsymbol{x}, \boldsymbol{x}') = k_1(\boldsymbol{x}_1, \boldsymbol{x}_1')k_2(\boldsymbol{x}_2, \boldsymbol{x}_2')$ is a covariance function.

Stationary Covariance Function and Isotropic Covariance Function

For simplicity of modeling of covariance functions, two classes of covariance functions are frequently discussed. One is the stationary covariance function and the other is the isotropic covariance function.

A stationary function is a function of $\boldsymbol{x} - \boldsymbol{x}'$ which is invariant to translation in the input space indicating correlation depends on the distance and direction of two location $\boldsymbol{x}$ and $\boldsymbol{x}'$. Moreover, a more simplified function is proposed as an isotropic function which only depends on the distance between two inputs $|\boldsymbol{x} - \boldsymbol{x}'|$. Therefore, if a function is stationary and semi-positive definite, it is a stationary covariance function. And if a function is isotropic and semi-positive definite, it is an isotropic covariance function.

Bochner gives a sufficient and necessary condition to construct stationary covariance function (Folland, 2016). It states that the covariance function of a stationary process can be represented as the Fourier transform of a positive finite measure.

Theorem 3.1.1. A complex-valued function k on $\mathbb{R}^D$ is the covariance function of a weakly stationary mean square continuous complex valued random process

on $\mathbb{R}^D$ if and only if it can be represented as

$$k(\boldsymbol{\gamma}) = \int_{\mathbb{R}^D} e^{2\pi i \boldsymbol{s}^T \boldsymbol{\gamma}} d\mu(\boldsymbol{s}) \tag{3.15}$$

where μ is a positive finite measure.

If μ has a density $S(\boldsymbol{s})$ then S is named as spectral density or power spectrum corresponding to kernel k. If the spectral density exists, there is a one to one mapping from covariance function and spectral density known as Wiener-Khintchine theorem.

$$k(\boldsymbol{\gamma}) = \int S(\boldsymbol{s}) e^{2\pi i \boldsymbol{s}^T \boldsymbol{\gamma}} d\boldsymbol{s}, S(\boldsymbol{s}) = \int k(\boldsymbol{\gamma}) e^{-2\pi i \boldsymbol{s}^T \boldsymbol{\gamma}} d\boldsymbol{\gamma}\,. \tag{3.16}$$

If the covariance function is isotropic, that implies it is a function of $r = |\boldsymbol{\gamma}|$. Then spectral density is a function of $s = |\boldsymbol{s}|$. Then

$$k(r) = \frac{2\pi}{r^{D/2-1}} \int_0^\infty S(s) J_{D/2-1}(2\pi r s) s^{D/2} ds\,, \tag{3.17}$$

$$S(s) = \frac{2\pi}{s^{D/2-1}} \int_0^\infty k(r) J_{D/2-1}(2\pi r s) r^{D/2} dr\,, \tag{3.18}$$

where $J_{D/2}$ is a Bessel function of order $D/2-1$. However the spectral density does not always exist and it depends on D. A necessary condition for the existence is that $\int r^{D-1}|k(r)|dr < \infty$.

Bochner's theorem is widely used to prove the positive definiteness of many usual stationary kernels.

- Gaussian kernels are the Fourier transform of itself.
- Matern kernels are the Fourier transforms of $\frac{1}{(1+w^2)^p}$.
- constant function is the Fourier transform of $\delta_{x,y}$

Bochner's theorem is the foundation of sparse spectrum Gaussian process (Lázaro-Gredilla et al., 2010). It sparsifies the power spectral density to obtain a sparse Gaussian process which is used to approximate the full Gaussian process.

Existing models for covariance function

This section would introduce some existing models for covariance function and their corresponding properties.

Squared Exponential Covariance Function

The squared exponential (SE) covariance function follows the form

$$k_{SE}(r) = \sigma^2 \exp\left(-\frac{r^2}{2\ell^2}\right) \tag{3.19}$$

with parameter σ^2 defining the characteristic scale and with parameter ℓ defining the characteristic length-scale. σ^2 determines the variability of functions from the mean and ℓ determines the smoothness of functions. Moreover, as ℓ increase, functions become more smooth, close to a linear function while when ℓ is small, functions are more like a random walk.

On the other hand, this covariance function is infinitely differentiable, implying that it has mean square derivatives of all orders. Therefore, it is very smooth. The smoothness contributes to a better inference under gradient-based optimization framework.

In the SE covariance function setting, $r = |\boldsymbol{x} - \boldsymbol{x}'|$. Usually the Euclidean distance is utilized, suggesting $r = \|\boldsymbol{x} - \boldsymbol{x}'\|_2$ where $\|\cdot\|_2$ denotes l_2 norm. However, since different components of input may have different contributions to the output of GP. Weighted Euclidean distance is considered to model the importance of different components of inputs. Therefore the distribution is redefined as $r = (\sum_{d=1}^{D} w_d(x_d - x'_d)^2)^{\frac{1}{2}}$. This approach is also named as automatic relevance determination (ARD) (MacKay, 1992; Neal, 1996). The components with large

weight are important while the components with small weight are not significantly important. Therefore, given the weight information, it is allowable to carry out model selection. A new model can be selected by using the s components of input with first s largest weights. ARD approach is very popular in machine learning, especially dealing with high dimensional dataset.

Matérn Class of Covariance function

The Matérn class of covariance functions is given by

$$k_{Matérn}(r) = \frac{2^{1-\nu}}{\Gamma(\nu)} \left(\frac{\sqrt{2\nu} r}{\ell} \right)^{\nu} K_{\nu} \left(\frac{\sqrt{2\nu} r}{\ell} \right) , \tag{3.20}$$

with positive parameters ν and ℓ, where K_{ν} is a modified Bessel function. As $\nu \to \infty$, this function degenerates to SE covariance function. In other words, the SE covariance function belongs to the Matérn class of covariance functions. In the Matérn class, a process $f(\boldsymbol{x})$ is k-times MS differentiable if and only if $\nu > k$. This covariance function is widely used in spatial statistics due to its flexibility and its robustness for numerical computation.

Periodic Covariance Functions

Periodic covariance functions are considered when observations have a significant periodicity. The idea of its construction comes from mapping its one dimensional input onto a circle on a two-dimensional space and then utilizing standard covariance function on the circle. Mathematically, we have a polar mapping $\boldsymbol{x}(t) = \begin{pmatrix} \cos(w_0 t) \\ \sin(w_0 t) \end{pmatrix}$. Then utilizing SE covariance function (3.19) on $\boldsymbol{x}$, we

have that

$$\begin{aligned} k(t,t') &= k_{SE}(\boldsymbol{x}(t),\boldsymbol{x}(t')) \\ &= \sigma^2 \exp\left(\frac{\|\boldsymbol{x}-\boldsymbol{x}'\|^2}{\ell}\right) \\ &= \sigma^2 \exp\left(-\frac{2\sin^2(w_0\frac{t-t'}{2})}{\ell^2}\right) \end{aligned} \tag{3.21}$$

In the periodic covariance function, w_0 fully specifies the periodic information and the period is equal to $\frac{2\pi}{w_0}$. Since the period information can only specified on $\mathbb{R}^1$, it is necessary to note that periodic covariance function only refers to GP with one dimensional input.

Nugget effects

Nugget effects initially come from the geo-science. It plays a very important role in kriging analysis. Nugget effects are defined as the sum of geological micro-structure and measurement errors (Haining, 1993). It is equivalent to the measurement errors in our case. For any covariance function k, the covariance function with nugget effects is given as

$$k_{NE}(\boldsymbol{x},\boldsymbol{x}') = k(\boldsymbol{x},\boldsymbol{x}') + \sigma^2\delta_{\boldsymbol{x},\boldsymbol{x}'}\,, \tag{3.22}$$

where δ is a Kronecker delta function. It indicates that all observations are corrupted by a white noise with variance σ^2. In general, nugget effects contribute to solving the over-fitting issue in Gaussian processes and improve the prediction accuracy.

Dimensionality Reduction

Gaussian processes are also used for dimensionality reduction and unsupervised learning, especially for high-dimensional data where the dimension of output D is greater than the number of data N.

Considering observed data $Y \in \mathbb{R}^{N \times D}$ where N is the number of observations and D is the dimension of output, those data are associated with latent variables $X \in \mathbb{R}^{N \times Q}$. In order to reduce the data dimension, latent variable dimension$Q$ should be set to much smaller than original data dimension D.

There exists many approaches to model the relation between X and Y in the latent variable model literature. Principle component analysis (PCA) is one of most popular dimensionality reduction methodologies. It maps the original data from high dimensionality to a lower dimensional sub-space. The principle components are recursively selected by the linear combination of observed variables which has the maximum variability under the complimentary sub-space with respect to the space generated by previous principle components. On the other hand, factor analysis is exactly in the opposite direction from PCA. It recursively considers a shared latent factor which causes the responds each component of observed variables. PPCA Tipping and Bishop (1999) reestablish the linkage between PCA and factor analysis and propose a linear embedding method for dimensionality reduction with a EM inference algorithm. The model is expressed as

$$\boldsymbol{y} \sim \mathcal{N}(\boldsymbol{y}|W\boldsymbol{x} + \boldsymbol{\mu}, \sigma^2 \boldsymbol{I}) \,, \tag{3.23}$$

with a standard Gaussian prior on latent variables $\boldsymbol{x} \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{I})$.

After marginalization with respect to $\boldsymbol{x}$, it has

$$\boldsymbol{y} \sim \mathcal{N}(\boldsymbol{y}|\boldsymbol{\mu}, C) \,, \tag{3.24}$$

where $C = WW^T + \sigma^2 \boldsymbol{I}$.

The corresponding log likelihood is

$$\ell = -\frac{N}{2}(D\ln(e\pi) + \ln|C| + \mathrm{tr}(C^{-1}S) \tag{3.25}$$

where

$$S = \frac{1}{N}\sum_{n=1}^{N}(\boldsymbol{y}_n - \boldsymbol{\mu})(\boldsymbol{y}_n - \boldsymbol{\mu})^T . \tag{3.26}$$

The maximum-likelihood estimators for parameter mean $\boldsymbol{\mu}$, weight matrix W and variance estimator σ^2 are

$$\boldsymbol{\mu}_{ML} = \frac{1}{N}\sum_{n=1}^{N}\boldsymbol{y}_n \tag{3.27}$$

$$W_{ML} = U_Q(\Lambda_Q - \sigma^2\boldsymbol{I})^{1/2}R \tag{3.28}$$

$$\sigma^2_{ML} = \frac{1}{D-Q}\sum_{j=Q+1}^{D}\lambda_j \tag{3.29}$$

where U_Q is the Q eigenvectors of S with corresponding diagonalized eigenvalues matrix $\Lambda_Q = \mathrm{diag}(\lambda_1, \ldots, \lambda_Q)$. And R is any $Q \times Q$ orthogonal rotation matrix.

To get out of the linear embedding restriction, kernel PCA is proposed by Schölkopf in Bernhard et al. (1998). It has a nonlinear transformation $\phi(\cdot)$ from the original D-dimensional feature space to a M-dimensional feature space where $M \gg D$, where the transformed features are assumed with zero mean. Then standard PCA algorithm is applied to the M-dimensional transformed data. Because of the nonlinear transformation, the embedding function is nonlinear since the embedding variables are expressed as a linear combination of transformed features. The kernel is expressed as $K(\boldsymbol{x}_i, \boldsymbol{x}_j) = \phi(\boldsymbol{x}_i)^T\phi(\boldsymbol{x}_j)$ and the popular kernels are enumerated as follows:

- Gaussian $K(\boldsymbol{x}_1, \boldsymbol{x}_2) = \exp(-\beta\|\boldsymbol{x}_1 - \boldsymbol{x}_2\|^2)$,

- Polynomial $K(\boldsymbol{x}_1, \boldsymbol{x}_2) = (1 + \boldsymbol{x}_1^T \boldsymbol{x}_2)^p$,
- Hyperbolic tangent $K(\boldsymbol{x}_1, \boldsymbol{x}_2) = \tanh(\boldsymbol{x}_1^T \boldsymbol{x}_2 + \delta)$.

However, there is no direct mapping from the embedded space to data-space which is named the pre-image problem. To reconstruct the original data, approximate methods are proposed for the pre-image problem in Wang (2012).

Other dimensionality reduction methods include multidimensional scaling, density network, spectral clustering and so on. All above methods focus on different parametric models between embedding inputs and observed data. The latent Gaussian process introduces a nonparametric approach to model the relation between embedding inputs and observed data. It is first mentioned by Lawrence (2004) and the model is named as GPLVM. It is a variant of probabilistic PCA. Instead of specifying a prior on latent variables, it specifies a prior distribution, $p(W) = \prod_{d=1}^{D} \mathcal{N}(\boldsymbol{w}_d|\boldsymbol{0}, \sigma_w^2 \boldsymbol{I})$ where $\boldsymbol{w}_i$ is the ith row of the weight matrix W. Then integrating over W the corresponding log-likelihood is

$$\ell = \frac{DN}{2} \ln(2\pi) - \frac{D}{2} \ln |K| - \frac{1}{2} \mathrm{tr}(K^{-1} Y Y^T) \,, \tag{3.30}$$

where $K = \sigma_w^{-2} X X^T + \sigma^2 \boldsymbol{I}$.

The maximum likelihood estimator of X is derived as

$$X_{ML} = U_Q L V^T \,, \tag{3.31}$$

where U_Q are Q eigenvectors of YY^T with corresponding diagonalized eigenvalues matrix $\Lambda_Q = \mathrm{diag}(\lambda_1, \ldots, \lambda_Q)$ of YY^T, L is a $q \times q$ diagonal matrix with jth element $l_j = \left(\frac{\sigma_w^2 \lambda_j}{D} - \sigma_w^2 \sigma^2\right)^{\frac{1}{2}}$, and V is arbitrary $Q \times Q$ orthogonal rotation matrix.

Log likelihood (3.30) shows that it is a product of D independent Gaussian processes with kernel function that $k(X, X) = \sigma_w^{-2} X X^T + \sigma^2 \boldsymbol{I}$ for each dimension

of observed data Y.

Therefore, the GPLVM is proposed by a natural extension of the non-linearisation of the mapping from X to Y through a general covariance function $k(\cdot, \cdot)$. The popular RBF kernel based GPLVM is discussed in Lawrence (2004). The paper also provides a practical algorithm for inference in which it recursively choose active set using IVM algorithm, update hyper-parameters and update embedding inputs sequentially.

Variational inference approach is applied to GPLVM in Titsias and Lawrence (2010). It integrates out the latent variables which appear nonlinearly in the inverse covariance matrix of GPLVM. Details are available in Section 3.1.6.

3.1.2 Low Rank Approximation based Sparse Gaussian Processes

In order to reduce the computational burden in (3.11), approximate Gaussian processes get increasing attention. Sparse Gaussian processes are one class of approximate Gaussian process models. Under the sparse Gaussian process setting, we consider all inputs and outputs as scalars for simplicity. Then we introduce inducing inputs $\boldsymbol{z} \in \mathbb{R}^M$ and inducing variables/output $\boldsymbol{u} \in \mathbb{R}^M$. Considering training latent variables $\boldsymbol{f}$ and test latent variables $\boldsymbol{f}^*$, the joint Gaussian priors and likelihood function is given as

$$p(\boldsymbol{f}, \boldsymbol{f}^*) = \mathcal{N}\left(\begin{pmatrix} \boldsymbol{f} \\ \boldsymbol{f}^* \end{pmatrix} \middle| \mathbf{0}, \begin{pmatrix} K_{\boldsymbol{x},\boldsymbol{x}} & K_{\boldsymbol{x},\boldsymbol{x}^*} \\ K_{\boldsymbol{x}^*,\boldsymbol{x}} & K_{\boldsymbol{x}^*,\boldsymbol{x}^*} \end{pmatrix}\right) \tag{3.32}$$

$$p(\boldsymbol{y}|\boldsymbol{f}) = \mathcal{N}(\boldsymbol{y}|\boldsymbol{f}, \sigma^2 \boldsymbol{I})\,. \tag{3.33}$$

It shows that the computation in (3.11) comes from the computation in the joint Gaussian priors. The fundamental approximation is to approximate the joint

prior by assuming testing variables and training variables are conditionally independent given $\boldsymbol{u}$, proposed in Quiñonero-Candela and Rasmussen (2005), implying that

$$p(\boldsymbol{f}, \boldsymbol{f}^*) \simeq q(\boldsymbol{f}, \boldsymbol{f}^*) = \int q(\boldsymbol{f}|\boldsymbol{u}) q(\boldsymbol{f}^*|\boldsymbol{u}) p(\boldsymbol{u}) d\boldsymbol{u}\,. \tag{3.34}$$

The true conditional distributions are

$$\begin{aligned} p(\boldsymbol{f}|\boldsymbol{u}) &= \mathcal{N}(\boldsymbol{f}|K_{\boldsymbol{x},\boldsymbol{z}}K_{\boldsymbol{z},\boldsymbol{z}}^{-1}\boldsymbol{u}, K_{x,x} - K_{\boldsymbol{x},\boldsymbol{z}}K_{\boldsymbol{z},\boldsymbol{z}}^{-1}K_{\boldsymbol{z},\boldsymbol{x}})\,, & (3.35)\\ p(\boldsymbol{f}^*|\boldsymbol{u}) &= \mathcal{N}(\boldsymbol{f}^*|K_{\boldsymbol{x}^*,\boldsymbol{z}}K_{\boldsymbol{z},\boldsymbol{z}}^{-1}\boldsymbol{u}, K_{x^*,x^*} - K_{\boldsymbol{x}^*,\boldsymbol{z}}K_{\boldsymbol{z},\boldsymbol{z}}^{-1}K_{\boldsymbol{z},\boldsymbol{x}^*})\,. & (3.36) \end{aligned}$$

Next, several approximation methods to approximate the conditional distributions are discussed in following sections.

Subset of Regression Approximation

The subset of regression (SoR) approximation considers a deterministic relation between training latent variables $\boldsymbol{f}$ and inducing variables $\boldsymbol{u}$ and a deterministic relation between testing latent variables $\boldsymbol{f}^*$ and inducing variables $\boldsymbol{u}$. The approximate condition distributions are given by:

$$\begin{aligned} \boldsymbol{f}_{SoR} &= K_{\boldsymbol{x},\boldsymbol{z}}K_{\boldsymbol{z},\boldsymbol{z}}^{-1}\boldsymbol{u}\,, & (3.37)\\ \boldsymbol{f}_{SoR}^* &= K_{\boldsymbol{x}^*,\boldsymbol{z}}K_{\boldsymbol{z},\boldsymbol{z}}^{-1}\boldsymbol{u} & (3.38) \end{aligned}$$

Because inducing variables have a Gaussian prior $\boldsymbol{u} \sim \mathcal{N}(\boldsymbol{0}, K_{\boldsymbol{z},\boldsymbol{z}})$, plugging approximate condition distributions (3.38) into Sparse Gaussian distribution as-

sumptions (3.34). The approximate joint distribution is given by

$$q_{SoR}(\boldsymbol{f}, \boldsymbol{f}^*) = \mathcal{N}\left(\begin{pmatrix}\boldsymbol{f}\\ \boldsymbol{f}^*\end{pmatrix}\middle|\boldsymbol{0}, \begin{pmatrix}Q_{\boldsymbol{x},\boldsymbol{x}} & Q_{\boldsymbol{x},\boldsymbol{x}^*}\\ Q_{\boldsymbol{x}^*,\boldsymbol{x}} & Q_{\boldsymbol{x}^*,\boldsymbol{x}^*}\end{pmatrix}\right) \tag{3.39}$$

where $Q_{\boldsymbol{x}_1,\boldsymbol{x}_2} = K_{\boldsymbol{x}_1,\boldsymbol{z}}K_{\boldsymbol{z},\boldsymbol{z}}^{-1}K_{\boldsymbol{z},\boldsymbol{x}_2}$. With the approximate joint distribution, the predictive distribution can be computed as

$$q_{SoR}(\boldsymbol{f}^*|\boldsymbol{y}) = \mathcal{N}(\boldsymbol{f}^*|Q_{\boldsymbol{x}^*,\boldsymbol{x}}(Q_{\boldsymbol{x},\boldsymbol{x}} + \sigma^2\boldsymbol{I})^{-1}\boldsymbol{y}, Q_{\boldsymbol{x}^*\boldsymbol{x}^*} - Q_{\boldsymbol{x}^*,\boldsymbol{x}}(Q_{\boldsymbol{x},\boldsymbol{x}} + \sigma^2\boldsymbol{I})^{-1}Q_{\boldsymbol{x},\boldsymbol{x}'}) \tag{3.40}$$

This predictive distribution is exactly the full GP predictive distribution (3.11) but covariance matrices K are replaced by approximate covariance matrices Q. Due to the Woodbury Matrix Identity, the computational complexity of $(Q_{\boldsymbol{x},\boldsymbol{x}} + \sigma^2\boldsymbol{I})^{-1}$ is $O(NM^2)$. Then we can derive that the computational complexity of the prediction distribution is $O(NM^2)$.

Deterministic Training Condition Approximation

By only considering a deterministic training condition in $q(\boldsymbol{f}|\boldsymbol{u})$, we achieve the deterministic training condition (DTC) approximation. Approximate conditional distributions are:

$$\begin{aligned} \boldsymbol{f}_{DTC} &= K_{\boldsymbol{x},\boldsymbol{z}}K_{\boldsymbol{z},\boldsymbol{z}}^{-1}\boldsymbol{u}\,, & (3.41)\\ q_{DTC}(\boldsymbol{f}^*|\boldsymbol{u}) &= p(\boldsymbol{f}^*|\boldsymbol{u}) \\ &= \mathcal{N}(\boldsymbol{f}^*|K_{\boldsymbol{x}^*,\boldsymbol{z}}K_{\boldsymbol{z},\boldsymbol{z}}^{-1}\boldsymbol{u}, K_{x^*,x^*} - Q_{x^*,x^*})\,. & (3.42)\end{aligned}$$

Then training conditional distribution is based on a projection $K_{\boldsymbol{x},\boldsymbol{z}}K_{\boldsymbol{z},\boldsymbol{z}}^{-1}\boldsymbol{u}$ where $K_{\boldsymbol{x},\boldsymbol{z}}K_{\boldsymbol{z},\boldsymbol{z}}^{-1}$ projects $\boldsymbol{u}$ to the conditional expectation $E[\boldsymbol{f}|\boldsymbol{u}]$. The conditional

marginal likelihood approximation is expressed as

$$p(\boldsymbol{y}|\boldsymbol{u}) \simeq q_{DTC}(\boldsymbol{y}|\boldsymbol{u}) = \mathcal{N}(\boldsymbol{y}|K_{\boldsymbol{x},\boldsymbol{z}}K_{\boldsymbol{z},\boldsymbol{z}}^{-1}\boldsymbol{u}, \sigma^2\boldsymbol{I}) \tag{3.43}$$

This approach is called Projected Latent Variable (PLV) by Seeger et al (Seeger, 2003). Plugging (3.42) to (3.34), the joint prior is approximated as

$$q_{DTC}(\boldsymbol{f}, \boldsymbol{f}^*) = \mathcal{N}\left(\begin{pmatrix}\boldsymbol{f}\\ \boldsymbol{f}^*\end{pmatrix}\middle|\boldsymbol{0}, \begin{pmatrix}Q_{\boldsymbol{x},\boldsymbol{x}} & Q_{\boldsymbol{x},\boldsymbol{x}^*}\\ Q_{\boldsymbol{x}^*,\boldsymbol{x}} & K_{\boldsymbol{x}^*,\boldsymbol{x}^*}\end{pmatrix}\right). \tag{3.44}$$

Then predictive distribution is

$$q_{DTC}(\boldsymbol{f}^*|\boldsymbol{y}) = \mathcal{N}(\boldsymbol{f}^*|Q_{\boldsymbol{x}^*,\boldsymbol{x}}(Q_{\boldsymbol{x},\boldsymbol{x}} + \sigma^2\boldsymbol{I})^{-1}\boldsymbol{y}, K_{\boldsymbol{x}^*\boldsymbol{x}^*} - Q_{\boldsymbol{x}^*,\boldsymbol{x}}(Q_{\boldsymbol{x},\boldsymbol{x}} + \sigma^2\boldsymbol{I})^{-1}Q_{\boldsymbol{x},\boldsymbol{x}^*}) \tag{3.45}$$

Predictive means under SoR and DTC are same but predictive covariance matrices are different.

Fully Independent (Training) Conditional Approximation

Instead of assuming a determinant relation between $\boldsymbol{f}$ adn $\boldsymbol{u}$, Fully independent training conditional (FITC) approximation method assumes the independence among training data. This approach is extended to sparse pseudo-input Gaussian processes (SPGPs) which is treated as a modified FITC in Snelson and Ghahramani (2006a).

As for FITC, approximate condition distributions are given as

$$\begin{aligned} q_{FITC}(\boldsymbol{f}|\boldsymbol{u}) &= \prod_{n=1}^{N} p(f_n|\boldsymbol{u}) \\ &= \mathcal{N}(\boldsymbol{f}|K_{\boldsymbol{x},\boldsymbol{z}}K_{\boldsymbol{z},\boldsymbol{z}}^{-1}\boldsymbol{u}, \mathrm{diag}(K_{\boldsymbol{x},\boldsymbol{x}} - Q_{\boldsymbol{x},\boldsymbol{x}}))\,, \qquad (3.46) \\ q_{FITC}(\boldsymbol{f}^*|\boldsymbol{u}) &= p(\boldsymbol{f}^*|\boldsymbol{u}) \\ &= \mathcal{N}(\boldsymbol{f}^*|K_{\boldsymbol{x}^*,\boldsymbol{z}}K_{\boldsymbol{z},\boldsymbol{z}}^{-1}\boldsymbol{u}, K_{\boldsymbol{x}^*,\boldsymbol{x}^*} - Q_{\boldsymbol{x}^*,\boldsymbol{x}^*})\,. \qquad (3.47) \end{aligned}$$

Comparing (3.42) and (3.47) with the true conditional distribution (3.36), Both DTC and FITC ignore the cross-correlation. On the other hand, FITC method remains the true variances on the diagonal of the covariance matrix while DTC method does not.

The joint prior is approximated as

$$q_{FITC}(\boldsymbol{f}, \boldsymbol{f}^*) = \mathcal{N}\left(\begin{pmatrix}\boldsymbol{f}\\ \boldsymbol{f}^*\end{pmatrix}\middle|\boldsymbol{0}, \begin{pmatrix} Q_{\boldsymbol{x},\boldsymbol{x}} + \mathrm{diag}(K_{\boldsymbol{x},\boldsymbol{x}} - Q_{\boldsymbol{x},\boldsymbol{x}}) & Q_{\boldsymbol{x},\boldsymbol{x}^*} \\ Q_{\boldsymbol{x}^*,\boldsymbol{x}} & K_{\boldsymbol{x}^*,\boldsymbol{x}^*}\end{pmatrix}\right) \qquad (3.48)$$

Letting $\tilde{Q}_{\boldsymbol{x},\boldsymbol{x}} = Q_{\boldsymbol{x},\boldsymbol{x}} + \mathrm{diag}(K_{\boldsymbol{x},\boldsymbol{x}} - Q_{\boldsymbol{x},\boldsymbol{x}})$, the predictive distribution is

$$q_{FITC}(\boldsymbol{f}^*|\boldsymbol{y}) = \mathcal{N}(\boldsymbol{f}^*|Q_{\boldsymbol{x}^*,\boldsymbol{x}}(\tilde{Q}_{\boldsymbol{x},\boldsymbol{x}} + \sigma^2\boldsymbol{I})^{-1}\boldsymbol{y}, K_{\boldsymbol{x}^*\boldsymbol{x}^*} - Q_{\boldsymbol{x}^*,\boldsymbol{x}}(\tilde{Q}_{\boldsymbol{x},\boldsymbol{x}} + \sigma^2\boldsymbol{I})^{-1}Q_{\boldsymbol{x},\boldsymbol{x}^*})\,. \qquad (3.49)$$

The posterior mean is rewritten as

$$\mu^* = K_{*,\boldsymbol{z}}\boldsymbol{\alpha} \qquad (3.50)$$

which implies that the posterior function can be expressed as a sum of basis functions $K_{*,\boldsymbol{z}}$ centering at the inducing inputs $\boldsymbol{z}$.

If we consider both fully independent training and testing condition, the joint

prior is approximated as

$$q_{FIC}(\boldsymbol{f}, \boldsymbol{f}^*) = \mathcal{N}\left(\begin{pmatrix}\boldsymbol{f}\\ \boldsymbol{f}^*\end{pmatrix}\middle|\boldsymbol{0}, \begin{pmatrix} Q_{\boldsymbol{x},\boldsymbol{x}} + \mathrm{diag}(K_{\boldsymbol{x},\boldsymbol{x}} - Q_{\boldsymbol{x},\boldsymbol{x}}) & Q_{\boldsymbol{x},\boldsymbol{x}^*} \\ Q_{\boldsymbol{x}^*,\boldsymbol{x}} & Q_{\boldsymbol{x}^*,\boldsymbol{x}^*} + \mathrm{diag}(K_{\boldsymbol{x}^*,\boldsymbol{x}^*} - Q_{\boldsymbol{x}^*,\boldsymbol{x}^*}) \end{pmatrix}\right)$$

which is named fully independent conditional (FIC) method.

Partial Independent (Training) Conditional Approximation

Partial independent training conditional approximation (PITC) is a generalization of FITC. It assumes the conditional independence among groups of observations. The approximate conditional distributions are given as

$$\begin{aligned} q_{PITC}(\boldsymbol{f}|\boldsymbol{u}) &= \prod_{s=1}^{S} p(\boldsymbol{f}_s|\boldsymbol{u}) \\ &= \mathcal{N}(\boldsymbol{f}|K_{\boldsymbol{x},\boldsymbol{z}}K_{\boldsymbol{z},\boldsymbol{z}}^{-1}\boldsymbol{u}, \mathrm{blockdiag}(K_{\boldsymbol{x},\boldsymbol{x}} - Q_{\boldsymbol{x},\boldsymbol{x}}))\,, \qquad (3.51) \\ q_{PITC}(\boldsymbol{f}^*|\boldsymbol{u}) &= p(\boldsymbol{f}^*|\boldsymbol{u}) \\ &= \mathcal{N}(\boldsymbol{f}^*|K_{\boldsymbol{x}^*,\boldsymbol{z}}K_{\boldsymbol{z},\boldsymbol{z}}^{-1}\boldsymbol{u}, K_{\boldsymbol{x}^*,\boldsymbol{x}^*} - Q_{\boldsymbol{x}^*,\boldsymbol{x}^*})\,. \qquad (3.52) \end{aligned}$$

This approach comes from Bayesian Committee Machine (Tresp, 2000) which assumes the independence among disjoint subsets of observations. It employs the test inputs as the inducing inputs in sparse Gaussian processes framework. This idea is relevant to the transduction learning. The joint prior is approximated as

$$q_{PITC}(\boldsymbol{f}, \boldsymbol{f}^*) = \mathcal{N}\left(\begin{pmatrix}\boldsymbol{f}\\ \boldsymbol{f}^*\end{pmatrix}\middle|\boldsymbol{0}, \begin{pmatrix} Q_{\boldsymbol{x},\boldsymbol{x}} + \mathrm{blockdiag}(K_{\boldsymbol{x},\boldsymbol{x}} - Q_{\boldsymbol{x},\boldsymbol{x}}) & Q_{\boldsymbol{x},\boldsymbol{x}^*} \\ Q_{\boldsymbol{x}^*,\boldsymbol{x}} & K_{\boldsymbol{x}^*,\boldsymbol{x}^*} \end{pmatrix}\right)\,.$$

Letting $\tilde{Q}_{\boldsymbol{x},\boldsymbol{x}} = Q_{\boldsymbol{x},\boldsymbol{x}} + \mathrm{blockdiag}(K_{\boldsymbol{x},\boldsymbol{x}} - Q_{\boldsymbol{x},\boldsymbol{x}})$, predictive distribution is

$$q_{PITC}(\boldsymbol{f}^*|\boldsymbol{y}) = \mathcal{N}(\boldsymbol{f}^*|Q_{\boldsymbol{x}^*,\boldsymbol{x}}(\tilde{Q}_{\boldsymbol{x},\boldsymbol{x}} + \sigma^2\boldsymbol{I})^{-1}\boldsymbol{y}, K_{\boldsymbol{x}^*\boldsymbol{x}^*} - Q_{\boldsymbol{x}^*,\boldsymbol{x}}(\tilde{Q}_{\boldsymbol{x},\boldsymbol{x}} + \sigma^2\boldsymbol{I})^{-1}Q_{\boldsymbol{x},\boldsymbol{x}^*})\,. \qquad (3.53)$$

.

There is no requirement of size for those blocks. For simplicity, suppose we have the same size B and then the training costs $\mathcal{O}(N/B * B^3) = \mathcal{O}(NB^2)$.

Since the predictive distribution is written as a weight sum of basis functions centering at those inducing points, the performance between PITC and FI(T)C is similar. Snelson and Ghahramani (2007) proposes partial independent conditional approximation where it assumes that testing input is grouped with training group such that

$$q(\boldsymbol{f}, \boldsymbol{f}^*|\boldsymbol{u}) = p(\boldsymbol{f}_S, \boldsymbol{f}^*|\boldsymbol{u}) \prod_{s=1}^{S-1} p(\boldsymbol{f}_s|\boldsymbol{u}) \,.$$

In the such way, the predictive mean is rewritten as a weighted sum of basis functions centered at the M inducing points and at the training inputs in the corresponding block.

3.1.3 Hyper-parameter Optimization in Sparse Gaussian Processes

This section introduces a standard way to find the optimal hyper-parameters for sparse Gaussian processes. The standard way is to maximize the log marginal likelihood rather than the posterior distribution. Maximizing the posterior distribution as maximum a posteriori (MAP) estimation is likely to cause over-fitting issue mainly due to that latent inputs are not marginalized (Damianou et al., 2016).

We summarize the log marginal likelihood for above four methods as follows:

- **SoR**, **DTC**:

$$\log q_{SoR}(\boldsymbol{y}) = \log q_{DTC}(\boldsymbol{y}) = \log\left(\mathcal{N}(\boldsymbol{y}|\boldsymbol{0}, Q_{\boldsymbol{x},\boldsymbol{x}} + \sigma^2\boldsymbol{I})\right) \tag{3.54}$$

- **FITC**:

$$\log q_{FITC}(\boldsymbol{y}) = \log\left(\mathcal{N}(\boldsymbol{y}|\boldsymbol{0}, Q_{\boldsymbol{x},\boldsymbol{x}} + \mathrm{diag}(K_{\boldsymbol{x},\boldsymbol{x}} - Q_{\boldsymbol{x},\boldsymbol{x}}) + \sigma^2\boldsymbol{I})\right) \tag{3.55}$$

- **PITC**:

$$\log q_{PITC}(\boldsymbol{y}) = \log\left(\mathcal{N}(\boldsymbol{y}|\boldsymbol{0}, Q_{\boldsymbol{x},\boldsymbol{x}} + \mathrm{blockdiag}(K_{\boldsymbol{x},\boldsymbol{x}} - Q_{\boldsymbol{x},\boldsymbol{x}}) + \sigma^2\boldsymbol{I})\right) \tag{3.56}$$

Therefore, the hyper-parameter optimization suggests maximizing those log marginal likelihood with respects to all hyper-parameters including noise scale parameter σ^2 and all parameters in the covariance function denoted as $\boldsymbol{\theta}$.

The time complexity of (3.54) and (3.55) is $\mathcal{O}(NM^2)$ using the Woodbury matrix formula and the matrix determinant lemma. Generally, as for any diagonal matrix D, we have

$$\begin{aligned}
& \log(\mathcal{N}(\boldsymbol{y}|\boldsymbol{0}, Q_{\boldsymbol{x},\boldsymbol{x}} + D)) \\
= \; & -\frac{N}{2}\log(2\pi) - \frac{1}{2}\log|Q_{\boldsymbol{x},\boldsymbol{x}} + D| - \frac{1}{2}\boldsymbol{y}^T(Q_{\boldsymbol{x},\boldsymbol{x}} + D)^{-1}\boldsymbol{y} \\
= \; & -\frac{N}{2}\log(2\pi) - \frac{1}{2}\log\left(\det(K_{\boldsymbol{z},\boldsymbol{z}} + K_{\boldsymbol{z},\boldsymbol{x}}D^{-1}K_{\boldsymbol{x},\boldsymbol{z}})\det(D)\det(K_{\boldsymbol{z},\boldsymbol{z}}^{-1})\right) \\
& -\frac{1}{2}\boldsymbol{y}^T\left(D^{-1} - D^{-1}K_{\boldsymbol{x},\boldsymbol{z}}\left(K_{\boldsymbol{z},\boldsymbol{z}} + K_{\boldsymbol{z},\boldsymbol{x}}D^{-1}K_{\boldsymbol{x},\boldsymbol{z}}\right)^{-1}K_{\boldsymbol{z},\boldsymbol{x}}D^{-1}\right)\boldsymbol{y}
\end{aligned} \tag{3.57}$$

whose time complexity is $\mathcal{O}(NM^2)$.

3.1.4 Bayesian Sparse Gaussian Processes

Bayesian inference is initially introduced to sparse Gaussian processes in Banerjee et al. (2008) named predictive processes. Then the bias-adjusted predictive process is proposed in Finley et al. (2009) to complement the biased variances. This idea is equivalent to that in the SPGP (Snelson and Ghahramani, 2006a). Then predictive processes are formally introduced to large spatial datasets and the computational efficiency is studied in Eidsvik et al. (2012). Guhaniyogi et al. (2011) study the knot selection in predictive processes.

3.1.5 Variational Sparse Gaussian Processes

This section discusses how variational inference is applied to sparse Gaussian processes. First we introduce general variational inference methodology in Section 3.1.5. We introduce two popular approaches for variational inference. The first variational learning approach for sparse Gaussian process is introduced by Titsias (Titsias, 2009) and is discussed in Section 3.1.5. Section 3.1.5 discusses another version of variational learning for sparse Gaussian processes which reduce the computation complexity from $O(NM^2)$ to $O(M^3)$.

Variational Inference

This section introduces general variational inference framework. We discuss the mean-field approximation method for variational distributions and empirical settings for variational distributions.

Variational Inference Framework

Variational inference is one of approximation methodologies for posterior inference in statistical model. Consider a general setting with observations $\boldsymbol{y} =$

$(y_1, \ldots, y_n)$ and latent variables $\boldsymbol{z} = (z_1, \ldots, z_m)$. The joint distribution is

$$p(\boldsymbol{y}, \boldsymbol{z}) = p(\boldsymbol{z})p(\boldsymbol{y}|\boldsymbol{z})\,.$$

According to the Bayesian rule, the posterior of latent variables follows

$$p(\boldsymbol{z}|\boldsymbol{y}) \propto p(\boldsymbol{z})p(\boldsymbol{y}|\boldsymbol{z})\,.$$

In practice, it is intractable to find a conjugate prior $p(\boldsymbol{z})$ such that $p(\boldsymbol{z}|\boldsymbol{y})$ has a closed form. Therefore, the variational inference is proposed. It proposes a variational distribution $q(\boldsymbol{z})$ depending on parameters $\boldsymbol{\theta}$. Throughout minimizing the Kullback-Leibler divergence between variational distributions and true posterior distributions with respect to parameters in variational distribution, the parameters in variational distributions are estimated by

$$\hat{\boldsymbol{\theta}} = \arg\min_{\boldsymbol{\theta}} \mathrm{KL}(q(\boldsymbol{z})||p(\boldsymbol{z}|\boldsymbol{y}))\,. \tag{3.58}$$

This approach is equivalent to maximizing the evidence lower boundary (ELBO). The ELBO is a lower boundary of log likelihood derived by Jensen's inequality.

$$\begin{aligned}
\log(p(\boldsymbol{y})) &= \log \int p(\boldsymbol{y}|\boldsymbol{z})p(\boldsymbol{z})d\boldsymbol{z} \\
&= \log \int p(\boldsymbol{y}|\boldsymbol{z})\frac{p(\boldsymbol{z})}{q(\boldsymbol{z})}q(\boldsymbol{z})d\boldsymbol{z} \\
&\geq \int \log\left(\frac{p(\boldsymbol{y}|\boldsymbol{z})p(\boldsymbol{z})}{q(\boldsymbol{z})}\right)q(\boldsymbol{z})d\boldsymbol{z}
\end{aligned} \tag{3.59}$$

There are two common expressions for the ELBO as

$$\begin{aligned} ELBO &= \int \log(p(\boldsymbol{y}|\boldsymbol{z}))q(\boldsymbol{z})d\boldsymbol{z} - \mathrm{KL}(q(\boldsymbol{z})||p(\boldsymbol{z})) && (3.60) \\ &= \log(p(\boldsymbol{y})) - \mathrm{KL}(q(\boldsymbol{z})||p(\boldsymbol{z}|\boldsymbol{y})) && (3.61) \end{aligned}$$

Because $p(\boldsymbol{y})$ does not depend on $\boldsymbol{\theta}$, according to (3.61), maximizing ELBO is equivalent to minimizing $\mathrm{KL}(q(\boldsymbol{z})||p(\boldsymbol{z}|\boldsymbol{y}))$.

In practice, in order to have a concise expression of EBLO, the expression (3.60) is always adapted in variational inference.

Mean-Field Approximation

The last section introduced the variational inference framework. This section introduces one approach to construct the variational distributions, Mean-Field approximation.

Mean-Field approximation assumes all latent variables are independent in variational distributions, mathematically implying $q(\boldsymbol{z}) = \prod_{j=1}^{m} q(z_j)$. Under this assumption, the optimal variational distributions are accessible by maximizing the ELBO, which is expressed as

$$ELBO = \int \log\left(\frac{p(\boldsymbol{y},\boldsymbol{z})}{\prod_{j=1}^{m} q(z_j)}\right) \prod_{j=1}^{m} q(z_j) d\boldsymbol{z}\,.$$

Then to get the optimal $q(\boldsymbol{z})$ is equivalent to solving an optimization problem such that

$$\begin{aligned} \max_{q(\boldsymbol{z})} \quad & ELBO \\ \text{subject to} \quad & \int q(z_j)dz_j = 1, \forall j = 1, \ldots m\,. \end{aligned}$$

The corresponding Lagrange equation is

$$L \;=\; ELBO - \sum_{j=1}^{m} \lambda_j (\int q(z_j) dz_j - 1) \tag{3.62}$$

We decompose (3.62) into to components, one with function $q(z_k)$ and the other without $q(z_k)$. Then

$$L \;=\; \int \Big(E_{q_{-k}} \log(p(\boldsymbol{y}, \boldsymbol{z})) q(z_k) - \log(q(z_k)) q(z_k) - \lambda_k q(z_k) \Big) dz_k + B \tag{3.63}$$

where q_{-k} denotes all functions $\{q(z_j)\}$ except $q(z_k)$ and B denotes other terms which do not include $q(z_k)$.

Let the functional derivative $\frac{\partial}{\partial q(z_k)} L = 0$. According to the Euler-Lagrange equation, we have

$$\begin{aligned} & \frac{\partial}{\partial q(z_k)} \Big(E_{q_{-k}} \log(p(\boldsymbol{y}, \boldsymbol{z})) q(z_k) dz_k - \log(q(z_k)) q(z_k) - \lambda_k q(z_k) \Big) \\ = \;& E_{q_{-k}} \log(p(\boldsymbol{y}, \boldsymbol{z})) - \log(q(z_k)) - 1 - \lambda_k = 0 \,. \end{aligned}$$

Then it suggests that $q(z_k) \propto \exp(E_{q_{-k}} \log(p(\boldsymbol{y}, \boldsymbol{z})))$. This mean-field approximate can not guarantee the closed-form expression for variational distribution $q(z_k)$.

Empirical Setting

For the ease of computation, we usually employ the empirical setting for variational distributions. Under the ELBO expression (3.60), two computational issues exist. The first one refers to the computation of the integration of $\int \log(p(\boldsymbol{y}|\boldsymbol{z})) q(\boldsymbol{z}) d\boldsymbol{z}$. This term is the marginal log likelihood with respect to variational distribution $q(\boldsymbol{z})$. In the aspect of machine learning, it is called the reconstruction term describing the log likelihood under $q(\boldsymbol{z})$. The other is the KL divergence between

the prior and the variational distributions $\mathrm{KL}(q(\boldsymbol{z})||p(\boldsymbol{z}))$. It is called the regularization term which makes $q(\boldsymbol{z})$ close to its prior.

As for the reconstruction term, usually it is computationally intractable. Sometimes Monte Carlo computation is required when the model structure is not conjugate. But as for the regularization term, to simplify the computation, the empirical setting assumes that $q(\boldsymbol{z})$ have the same class of distribution of $q(\boldsymbol{z})$. Under this assumption, the KL divergence usually has a closed form expression.

In general, we take five classes of distributions for instance:

- KL divergence of inverse Gamma distributions: Assume $p \sim \mathcal{IG}(\alpha, \beta)$ and $q \sim \mathcal{IG}(\tilde{\alpha}, \tilde{\beta})$. Then KL divergence between p and q is expressed as:

$$\begin{aligned} \mathcal{KL}[p||q] &= \int_0^\infty p(x) \log \frac{p(x)}{q(x)} dx \\ &= (\alpha - \tilde{\alpha})\Psi(\alpha) + \tilde{\beta}(\frac{\alpha}{\beta}) - \alpha + \log \frac{\beta^{\tilde{\alpha}+1}\Gamma(\tilde{\alpha})}{\beta\tilde{\beta}^{\tilde{\alpha}}\Gamma(\alpha)} \,. \end{aligned}$$

- KL divergence of Dirichlet distributions: Assume $p \sim \mathrm{Dir}(\boldsymbol{c})$ and $q \sim \mathrm{Dir}(\tilde{\boldsymbol{c}})$. Then KL divergence between p and q is expressed as:

$$\mathcal{KL}[p||q] = \log\Gamma(c_0) - \sum_{i=1}^{n} \log\Gamma(c_i) - \log\Gamma(\tilde{c}_0) + \sum_{i=1}^{n} \log\Gamma(\tilde{c}_i) + \sum_{i=1}^{n} (c_i - \tilde{c}_i)(\Psi(c_i) - \Psi(c_0)) \,.$$

- KL divergence of Gaussian distributions: Assume $p \sim \mathcal{N}(\mu, \sigma^2)$ and $q \sim \mathcal{N}(\tilde{\mu}, \tilde{\sigma}^2)$. Then KL divergence between p and q is expressed as:

$$\mathcal{KL}[p||q] = \log \frac{\tilde{\sigma}}{\sigma} + \frac{\sigma^2 + (\mu - \tilde{\mu})^2}{2\tilde{\sigma}^2} - \frac{1}{2} \,.$$

- KL divergence of multivariate Gaussian distributions with D dimension: Assume $p \sim \mathcal{N}(\boldsymbol{\mu}, \Sigma)$ and $q \sim \mathcal{N}(\tilde{\boldsymbol{\mu}}, \tilde{\Sigma})$. Then KL divergence between p and

q is expressed as:

$$\mathcal{KL}[p||q] = \frac{1}{2}\left(\log\frac{|\tilde{\Sigma}|}{|\Sigma|} - D + \mathrm{tr}(\tilde{\Sigma}^{-1}\Sigma) + (\tilde{\boldsymbol{\mu}} - \boldsymbol{\mu})^T\tilde{\Sigma}^{-1}(\tilde{\boldsymbol{\mu}} - \boldsymbol{\mu})\right) . \quad (3.64)$$

- KL divergence of categorical distributions: Assume $p \sim \mathrm{Cat}(\boldsymbol{\pi})$ and $q \sim \mathrm{Cat}(\tilde{\boldsymbol{\pi}})$. Then KL divergence between p and q is expressed as:

$$\mathcal{KL}[p||q] = \sum_{i=1}^{n} \pi_i \log\frac{\pi_i}{\tilde{\pi}_i} .$$

Inference based on Marginalization

This section introduces one variational learning approach for sparse Gaussian process proposed by Titsias (Titsias, 2009), named as SGPR. Consider observations $\boldsymbol{y}$ with latent variables $\boldsymbol{f}$ and covariates $\boldsymbol{x}$ and consider inducing inputs $\boldsymbol{z}$ and latent inducing variables $\boldsymbol{f}_m$. The joint variational distribution of both $\boldsymbol{f}$ and $\boldsymbol{f}_m$ are proposed as follows:

$$q(\boldsymbol{f}, \boldsymbol{f}_m) = p(\boldsymbol{f}|\boldsymbol{f}_m)q(\boldsymbol{f}_m) .$$

The log likelihood has such a ELBO that

$$\begin{aligned}
\log p(\boldsymbol{y}) &= \log \int p(\boldsymbol{y}, \boldsymbol{f}, \boldsymbol{f}_m) d\boldsymbol{f} d\boldsymbol{f}_m \\
&= \log \int \frac{p(\boldsymbol{y}, \boldsymbol{f}, \boldsymbol{f}_m)}{q(\boldsymbol{f}, \boldsymbol{f}_m)} q(\boldsymbol{f}, \boldsymbol{f}_m) d\boldsymbol{f} d\boldsymbol{f}_m \\
&\geq \int \log \left(\frac{p(\boldsymbol{y}, \boldsymbol{f}, \boldsymbol{f}_m)}{q(\boldsymbol{f}, \boldsymbol{f}_m)} \right) q(\boldsymbol{f}, \boldsymbol{f}_m) d\boldsymbol{f} d\boldsymbol{f}_m \\
&= \int \log \left(\frac{p(\boldsymbol{y}|\boldsymbol{f}) p(\boldsymbol{f}|\boldsymbol{f}_m) p(\boldsymbol{f}_m)}{p(\boldsymbol{f}|\boldsymbol{f}_m) q(\boldsymbol{f}_m)} \right) q(\boldsymbol{f}, \boldsymbol{f}_m) d\boldsymbol{f} d\boldsymbol{f}_m \\
&= \int \log \left(\frac{p(\boldsymbol{y}|\boldsymbol{f}) p(\boldsymbol{f}_m)}{q(\boldsymbol{f}_m)} \right) q(\boldsymbol{f}, \boldsymbol{f}_m) d\boldsymbol{f} d\boldsymbol{f}_m \qquad (3.65) \\
&= \int \log(p(\boldsymbol{y}|\boldsymbol{f})) q(\boldsymbol{f}, \boldsymbol{f}_m) d\boldsymbol{f} d\boldsymbol{f}_m - \mathrm{KL}(q(\boldsymbol{f}_m)||p(\boldsymbol{f}_m)) \qquad (3.66)
\end{aligned}$$

To get the optimal variational distribution $q(\boldsymbol{f}_m)$, given (3.65), it is equivalent to solve the constrained optimization problem that

$$\begin{aligned}
\max_{q(\boldsymbol{f}_m)} \quad & \int \log \left(\frac{p(\boldsymbol{y}|\boldsymbol{f}) p(\boldsymbol{f}_m)}{q(\boldsymbol{f}_m)} \right) q(\boldsymbol{f}, \boldsymbol{f}_m) d\boldsymbol{f} d\boldsymbol{f}_m \\
\text{subject to} \quad & \int q(\boldsymbol{f}_m) d\boldsymbol{f}_m = 1 \,.
\end{aligned}$$

The corresponding Lagrange equation is

$$\begin{aligned}
L &= \int \log \left(\frac{p(\boldsymbol{y}|\boldsymbol{f}) p(\boldsymbol{f}_m)}{q(\boldsymbol{f}_m)} \right) q(\boldsymbol{f}, \boldsymbol{f}_m) d\boldsymbol{f} d\boldsymbol{f}_m - \lambda \left(\int q(\boldsymbol{f}_m) d\boldsymbol{f}_m - 1 \right) \\
&= \int \left(\int \log p(\boldsymbol{y}|\boldsymbol{f}) p(\boldsymbol{f}|\boldsymbol{f}_m) d\boldsymbol{f} + \log \left(\frac{p(\boldsymbol{f}_m)}{q(\boldsymbol{f}_m)} \right) - \lambda \right) q(\boldsymbol{f}_m) d\boldsymbol{f}_m + \lambda \,.
\end{aligned}$$

Letting $\frac{\partial L}{\partial q(\boldsymbol{f}_m)} = 0$, the Euler-Lagrange equation shows that

$$\int \log p(\boldsymbol{y}|\boldsymbol{f}) p(\boldsymbol{f}|\boldsymbol{f}_m) d\boldsymbol{f} + \log \left(\frac{p(\boldsymbol{f}_m)}{q(\boldsymbol{f}_m)} \right) - \lambda - 1 = 0$$

which implies

$$q(\boldsymbol{f}_m) \propto p(\boldsymbol{f}_m)\exp(\int \log p(\boldsymbol{y}|\boldsymbol{f})p(\boldsymbol{f}|\boldsymbol{f}_m)d\boldsymbol{f})\,, \tag{3.67}$$

where $p(\boldsymbol{f}|\boldsymbol{f}_m)=\mathcal{N}(\tilde{\boldsymbol{\mu}},\tilde{K})$, $\tilde{\boldsymbol{\mu}}=K_{\boldsymbol{x},\boldsymbol{z}}K_{\boldsymbol{z},\boldsymbol{z}}^{-1}\boldsymbol{f}_m$, $\tilde{K}=K_{\boldsymbol{x},\boldsymbol{x}}-K_{\boldsymbol{x},\boldsymbol{z}}K_{\boldsymbol{z},\boldsymbol{z}}^{-1}K_{\boldsymbol{z},\boldsymbol{x}}$.

On the other hand,

$$\begin{aligned}\int \log p(\boldsymbol{y}|\boldsymbol{f})p(\boldsymbol{f}|\boldsymbol{f}_m)d\boldsymbol{f} &= E_{p(\boldsymbol{f}|\boldsymbol{f}_m)}\log\mathcal{N}(\boldsymbol{y}|\boldsymbol{f},\sigma^2\boldsymbol{I})\\ &= E_{p(\boldsymbol{f}|\boldsymbol{f}_m)}\left(-\frac{n}{2}\log(2\pi\sigma^2)-\frac{1}{2\sigma^2}(\boldsymbol{y}^T\boldsymbol{y}-2\boldsymbol{f}^T\boldsymbol{y}+\boldsymbol{f}^T\boldsymbol{f})\right)\\ &= -\frac{n}{2}\log(2\pi\sigma^2)-\frac{1}{2\sigma^2}\mathrm{tr}\left(\boldsymbol{y}\boldsymbol{y}^T-2\boldsymbol{y}\tilde{\boldsymbol{\mu}}^T+\tilde{\boldsymbol{\mu}}\tilde{\boldsymbol{\mu}}^T+\tilde{K}\right)\\ &= \log\mathcal{N}(\boldsymbol{y}|\tilde{\boldsymbol{\mu}},\sigma^2\boldsymbol{I})-\frac{1}{2\sigma^2}\mathrm{tr}(\tilde{K})\,.\end{aligned} \tag{3.68}$$

Because $\tilde{\boldsymbol{\mu}}$ depends on $\boldsymbol{f}_m$ but $Q_{\boldsymbol{x},\boldsymbol{x}}$ does not depend on $\boldsymbol{f}_m$, we derive

$$\begin{aligned}q(\boldsymbol{f}_m) &\propto p(\boldsymbol{f}_m)\mathcal{N}(\boldsymbol{y}|\tilde{\boldsymbol{\mu}},\sigma^2\boldsymbol{I})\Rightarrow q(\boldsymbol{f}_m)\\ &= \frac{p(\boldsymbol{f}_m)\mathcal{N}(\boldsymbol{y}|\tilde{\boldsymbol{\mu}},\sigma^2\boldsymbol{I})}{\int p(\boldsymbol{f}_m)\mathcal{N}(\boldsymbol{y}|\tilde{\boldsymbol{\mu}},\sigma^2\boldsymbol{I})d\boldsymbol{f}_m}\\ &= \mathcal{N}(\boldsymbol{\mu},A)\end{aligned} \tag{3.69}$$

where $\boldsymbol{\mu}=\sigma^{-2}K_{\boldsymbol{z},\boldsymbol{z}}\Sigma K_{\boldsymbol{z},\boldsymbol{x}}\boldsymbol{y}$, $A=K_{\boldsymbol{z},\boldsymbol{z}}\Sigma K_{\boldsymbol{z},\boldsymbol{z}}$ and $\Sigma=(K_{\boldsymbol{z},\boldsymbol{z}}+\sigma^{-2}K_{\boldsymbol{z},\boldsymbol{x}}K_{\boldsymbol{x},\boldsymbol{z}})^{-1}$ as derived in Titsias (2009). Plugging (3.69) into (3.66), the ELBO is expressed as

$$\begin{aligned}ELBO &= \int\log\left(\frac{\mathcal{N}(\boldsymbol{y}|\tilde{\boldsymbol{\mu}},\sigma^2\boldsymbol{I})p(\boldsymbol{f}_m)}{q(\boldsymbol{f}_m)}\right)q(\boldsymbol{f}_m)d\boldsymbol{f}_m-\frac{1}{2\sigma^2}\mathrm{tr}(\tilde{K})\\ &= \log\int\mathcal{N}(\boldsymbol{y}|\tilde{\boldsymbol{\mu}},\sigma^2\boldsymbol{I})p(\boldsymbol{f}_m)d\boldsymbol{f}_m-\frac{1}{2\sigma^2}\mathrm{tr}(\tilde{K})\\ &= \log\mathcal{N}(\boldsymbol{y}|\boldsymbol{0},Q_{\boldsymbol{x},\boldsymbol{x}}+\sigma^2\boldsymbol{I})-\frac{1}{2\sigma^2}\mathrm{tr}(\tilde{K})\,.\end{aligned} \tag{3.70}$$

Comparing (3.70) with (3.54), (3.70) has one more term $-\frac{1}{2\sigma^2}\mathrm{tr}(\tilde{K})$. It illus-

trates this variational approach is different from SoR and DTC.

Next we derive the approximate posterior predictive distribution. As for new inputs $\boldsymbol{x}^*$, the posterior predictive distribution $\boldsymbol{f}^*$ is

$$\begin{aligned} q(\boldsymbol{f}^*) &= \int q(\boldsymbol{f}^*, \boldsymbol{f}_m) d\boldsymbol{f}_m \\ &= p(\boldsymbol{f}^*|\boldsymbol{f}_m) q(\boldsymbol{f}_m) d\boldsymbol{f}_m \\ &= \int \mathcal{N}(\boldsymbol{f}^*|K_{\boldsymbol{x}^*,\boldsymbol{z}} K_{\boldsymbol{z},\boldsymbol{z}}^{-1} \boldsymbol{f}_m, K_{\boldsymbol{x}^*,\boldsymbol{x}^*} - K_{\boldsymbol{x}^*,\boldsymbol{z}} K_{\boldsymbol{z},\boldsymbol{z}}^{-1} K_{\boldsymbol{z},\boldsymbol{x}^*}) \mathcal{N}(\boldsymbol{f}_m|\boldsymbol{\mu}, A) d\boldsymbol{f}_m \\ &= \mathcal{N}(\boldsymbol{f}^*|K_{\boldsymbol{x}^*,\boldsymbol{z}} K_{\boldsymbol{z},\boldsymbol{z}}^{-1} \boldsymbol{\mu}, K_{\boldsymbol{x}^*,\boldsymbol{x}^*} - K_{\boldsymbol{x}^*,\boldsymbol{z}} K_{\boldsymbol{z},\boldsymbol{z}}^{-1} K_{\boldsymbol{z},\boldsymbol{x}^*} + K_{\boldsymbol{x}^*,\boldsymbol{z}} B K_{\boldsymbol{z},\boldsymbol{x}^*}) \quad (3.71) \end{aligned}$$

where $B = K_{\boldsymbol{z},\boldsymbol{z}}^{-1} A K_{\boldsymbol{z},\boldsymbol{z}}^{-1}$.

Inference based on Variational Distribution

This section discusses an alternative variational learning approach named SVGP proposed in Hensman et al. (2013), which does not need to compute the optimal variational distribution in (3.69). It directly assumes that variational distribution follows a multivariate normal distribution $q(\boldsymbol{f}_m) = \mathcal{N}(\boldsymbol{f}_m|\boldsymbol{m}, S)$. Then under this assumption, given (3.66) and (3.68), the EBLO is derived as

$$\begin{aligned} ELBO &= \int \log(p(\boldsymbol{y}|\boldsymbol{f})) q(\boldsymbol{f}, \boldsymbol{f}_m) d\boldsymbol{f} d\boldsymbol{f}_m - \mathrm{KL}(q(\boldsymbol{f}_m)||p(\boldsymbol{f}_m)) \\ &= \int \left(\log \mathcal{N}(\boldsymbol{y}|\tilde{\boldsymbol{\mu}}, \sigma^2 \boldsymbol{I}) - \frac{1}{2\sigma^2} \mathrm{tr}(\tilde{K}) \right) q(\boldsymbol{f}_m) d\boldsymbol{f}_m - \mathrm{KL}(q(\boldsymbol{f}_m)||p(\boldsymbol{f}_m)) \\ &= E_{q(\boldsymbol{f}_m)} \left(\log \left(\mathcal{N}(\boldsymbol{y}|K_{\boldsymbol{x},\boldsymbol{z}} K_{\boldsymbol{z},\boldsymbol{z}}^{-1} \boldsymbol{f}_m, \sigma^2 \boldsymbol{I}) \right) \right) - \frac{1}{2\sigma^2} \mathrm{tr}(\tilde{K}) - \mathrm{KL}(q(\boldsymbol{f}_m)||p(\boldsymbol{f}_m)) \\ &= \sum_{i=1}^{n} \left(\log \left(\mathcal{N}(y_i|K_{\boldsymbol{x},\boldsymbol{z}} K_{\boldsymbol{z},\boldsymbol{z}}^{-1} \boldsymbol{m}, \sigma^2) \right) - \frac{1}{2\sigma^2} S K_{\boldsymbol{z},\boldsymbol{z}}^{-1} k_i k_i^T K_{\boldsymbol{z},\boldsymbol{z}}^{-1} - \frac{1}{2} \tilde{K}_{i,i} \right) \\ &\quad - \mathrm{KL}(q(\boldsymbol{f}_m)||p(\boldsymbol{f}_m)) \end{aligned} \quad (3.72)$$

The computation is decomposed for each observation y_i, where the stochastic variational inference is feasible. And the computational complexity for each ob-

servation is $O(M^3)$. As for the KL term, because of (3.64), the computational complexity is $O(M^3)$. Therefore, the total computational complexity can be scaled as $O(M^3)$.

The predictive distribution for a new input $\boldsymbol{x}^*$ is

$$\begin{aligned} q(\boldsymbol{f}^*) &= \int q(\boldsymbol{f}^*, \boldsymbol{f}_m) d\boldsymbol{f}_m \\ &= \mathcal{N}(\boldsymbol{f}^* | K_{\boldsymbol{x}^*,\boldsymbol{z}} K_{\boldsymbol{z},\boldsymbol{z}}^{-1} \boldsymbol{m}, K_{\boldsymbol{x}^*,\boldsymbol{x}^*} - K_{\boldsymbol{x}^*,\boldsymbol{z}} K_{\boldsymbol{z},\boldsymbol{z}}^{-1} K_{\boldsymbol{z},\boldsymbol{x}^*} + K_{\boldsymbol{x}*,\boldsymbol{z}} K_{\boldsymbol{z},\boldsymbol{z}}^{-1} S K_{\boldsymbol{z},\boldsymbol{z}}^{-1} K_{\boldsymbol{z},\boldsymbol{x}^*}) \,. \end{aligned} \tag{3.73}$$

.

Considering the previous variational learning approach, the computational complexity of ELBO in (3.65) is $\mathcal{O}(NM^2)$. Therefore, this approach is much computationally cheaper than the last one, especially when the number of observations is large.

3.1.6 Experiments

This section applies different Gaussian processes models on a synthetic dataset and compares their inference results.

Synthetic data

We generate our data from a function such as

$$y = f(t) + 0.5\epsilon = \frac{\sin(2\pi t)}{2\pi t} + 0.5\epsilon \,, \tag{3.74}$$

where $\epsilon \sim \mathcal{N}(0, 1)$.

We take 2000 inputs from a standard normal $\mathcal{N}(0, 1)$ denoted as $\boldsymbol{t} = (t_1, \ldots, t_{2000})$ and generate corresponding observations $\boldsymbol{y} = (y_1, \ldots, y_{2000})$ according to the gen-

erator (3.74). And we randomly select 100 testing inputs from a standarad normal $\mathcal{N}(0,1)$ denoted as $\tilde{\boldsymbol{t}} = (\tilde{t}_1, \ldots, \tilde{t}_{100})$ with latent testing variables denoted as $\tilde{\boldsymbol{f}} = (\tilde{f}_1, \ldots, \tilde{f}_{100})$.

Different Gaussian process models are considered for inference. First we use a full Gaussian process model. Then we consider sparse Gaussian processes from a frequentist, Bayesian, and variational Bayesian perspectives respectively.

Frequentist Inference

From a frequentist perspective, we consider SoR, DTC, FITC three models with RFB covariance function. We choose 10 inducing inputs and the initial inducing inputs are assume evenly distributed on the interval $(-2, 2)$. After hyper-parameter optimization discussed in section 3.1.3. We estimate the optimal inducing inputs and hyper-parameters in covariance function. In order to compare the computation time, we use the same L-BFGS in optimization and set the maximum number of iterations as 10. With those estimates, we plot the posterior predictive processes in Figure 3.1. Using the posterior mean as predictive estimates, we compute the mean of square errors (MSE) and the mean of absolute differences (MAD) as two predictive criteria for testing variables $\tilde{\boldsymbol{f}}$. All results are shown in Table 3.1.

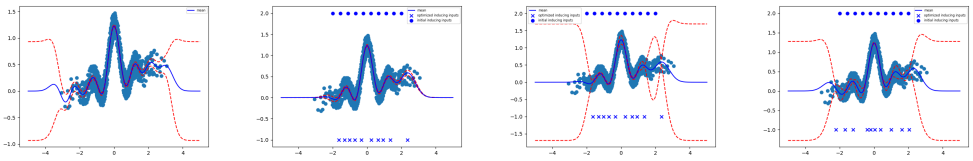

Figure 3.1: Posterior predictive processes for four models. From left to right, they are fully GP, SoR, DTC and FITC models. The blue line denotes the predictive mean. The red dashed lines denote the 95% credible intervals. The blue circles denote initial inducing points and the blue crosses denote optimized inducing inputs.

Table 3.1: Hyper-parameter optimization time and predictive accuracy for four different models: GP, SoR, DTC and FITC. Mean of square errors (MSE) and mean of absolute difference (MAD) are summarized for 100 testing data.

Model	GP	SoR	DTC	FITC
Time	17.04s	6.08s	6.46s	6.91s
MSE	0.060	0.062	0.062	0.064
MAD	0.244	0.248	0.248	0.250

Table 3.1 shows that SoR, DTC and FITC are much faster than a full Gaussian process model and have very competitive predictive results. And Figure 3.1 shows that FITC has the most similar uncertainty as the fully Gaussian processes model compared with SoR and DTC.

Bayesian Inference

As for Bayesian inference, we consider both predictive processes (PP) (Banerjee et al., 2008) and adjusted-bias predictive processes (APP) (Finley et al., 2009). We set 10 inducing inputs $\boldsymbol{Z} = (Z_1, \ldots, Z_{10})$ uniformly sampled on $(-2, 2)$ and set 2000 sampling iterations.

We set priors for hyper-parameters as

$$\begin{aligned} \sigma^2 &\sim \mathcal{IG}(2,1) \\ l &\sim \mathcal{IG}(2,1) \\ \sigma^2_{error} &\sim \mathcal{IG}(2,1) \end{aligned} \tag{3.75}$$

where σ^2, l are variance parameter and length-scale parameter in the RFB covariance function and σ^2_{noise} is the variance parameter of measure errors.

Before proceeding to the MCMC procedure, we compute the maximum likelihood estimate of $\boldsymbol{\theta} = (\log \sigma_2, \log l, \log \sigma^2_{error})$ denoted as $\hat{\boldsymbol{\theta}}$ and compute the observed fisher information matrix with respect to $\boldsymbol{\theta}$ by computing the negative

inverse Hessian matrix of log-likelihood at $\hat{\boldsymbol{\theta}}$. Then $\hat{\boldsymbol{\theta}}$ is utilized as initial starting points and the observed fisher information matrix is treated as the covariance matrix in transition kernel in the Metropolis Hasting approach. The details about advanced MCMC algorithm can be found in appendix A.1.

We record the time for the Metropolis Hasting algorithm. We compute predictive estimates using posterior means. The mean of square errors (MSE) and the mean of absolute differences (MAD) are calculated as two predictive criteria for testing variables $\tilde{\boldsymbol{f}}$. All results are shown in Table 3.2.

Table 3.2: Metropolis Hasting time and predictive accuracy for four different models: PP and APP. Mean of square errors (MSE) and mean of absolute difference (MAD) are summarized for 100 testing data.

Model	PP	APP
Time	186.15s	182.39s
MSE	0.061	0.062
MAD	0.246	0.247

Comparing Bayesian inference with Frequentist inference, Bayesian inference can provide sensitivity analysis for hyper-parameters but Frequnetist inference can not. The posterior distributions are provided in Figures 3.2. However, Bayesian inference is much more expensive than Frequentist inference. Next section discusses two variational inferences which directly approximates the posterior distribution of inducing variables by a variational distribution.

Variational Inference

This section discusses two variational approaches for the synthetic data. One is the SGPR model and the other is the SVGP model.

As for the SGPR model, through maximizing the ELBO, we optimize all hyper-parameters. We record this training time for 10 iterations in the L-BFGS algorithm. Then according to (3.71), we get the approximate posterior predictive

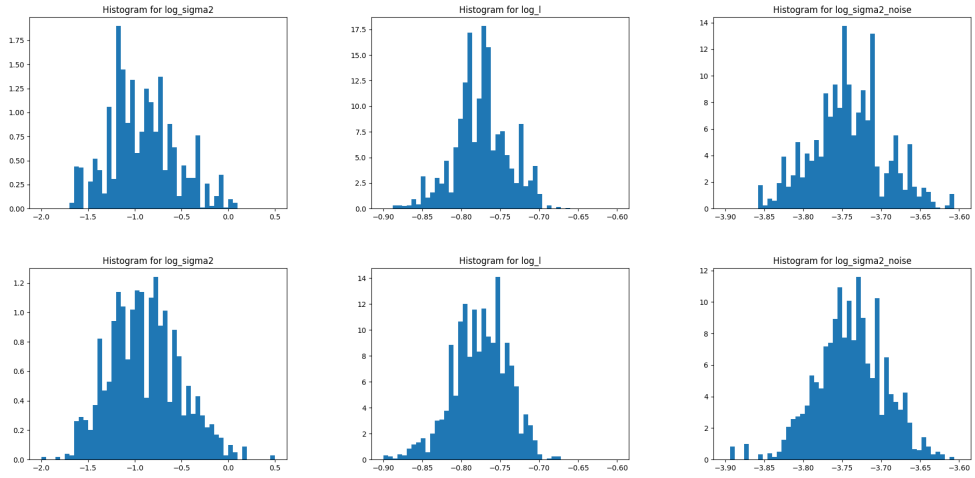

Figure 3.2: Posterior distribution of hyperparameters on log scale $\log \sigma^2, \log l$ and $\log \sigma^2_{error}$ for PP (upper) and APP (bottom) model separately.

process shown in Figure 3.3. The same two criteria, MSE and MAD, are calculated in Table 3.3.

As for the SVGP model, we set 2000 iterations in the stochastic gradient descent optimization. The training time and two criteria of predictive accuracy are summarized in Table 3.3. Although the SVGP result is little worse than the SGPR, it is much faster for each iteration in optimization.

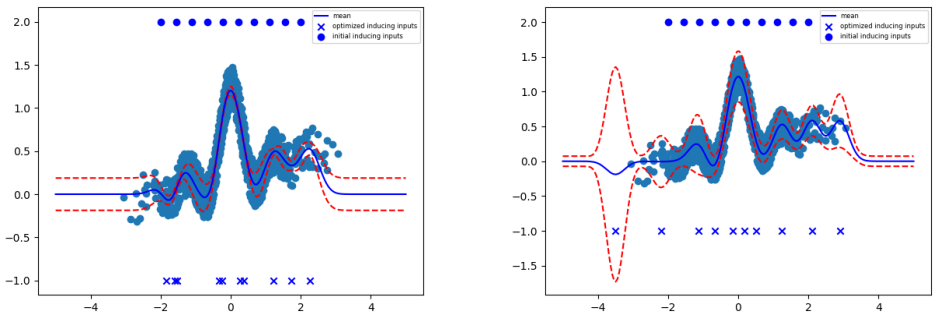

Figure 3.3: Posterior predictive processes for two models: SGPR model (left) and SVGP model (right). The blue line denotes predictive mean. The red dashed lines denote 95% credible interval. The blue circles denote initial inducing points and the blue crosses denote optimized inducing inputs.

Table 3.3: Training time and predictive accuracy for two different models: SGPR and SVGP. Mean of square errors (MSE) and mean of absolute difference (MAD) are summarized for 100 testing data.

Model	SGPR	SVGP
Time	6.89s	32.69s
MSE	0.056	0.059
MAD	0.214	0.242

Conclusion

Sparse Gaussian process approaches: SoR, DTC, FITC and SGPR have similar running time and prediction performance. Bayesian methods cost more time because of the requirement of sampling. And the SVGP approach is faster but it is not robust and strongly depends on the starting points.

3.2 Regularization for Sparse Gaussian Processes

In this section, we first discuss the motivation of our regularization approach in inducing-point based sparse GP models. Then we propose a non-parametric regularizer and a parametric regularizer respectively, and illustrate our regularization performance on 1D synthetic data.

We consider an naive regression model for illustration. Suppose we have observations $\boldsymbol{y} = (y_1, y_2, \ldots, y_N)$ with predictors $\boldsymbol{x} = (x_1, x_2, \ldots, x_N)$. A full Gaussian process model is

$$\begin{aligned} &y_n = f_n + \epsilon_n\,, \qquad \epsilon_n \overset{iid}{\sim} \mathcal{N}(\epsilon_n|0, \sigma_{err}^2)\,, \qquad f_n = f(x_n) \qquad \forall n = 1, 2, \ldots, N\,, \\ &f \sim \mathrm{GP}(0, C(\boldsymbol{\theta}))\,, \end{aligned} \tag{3.76}$$

where f is given a GP prior with a covariance function $C(\boldsymbol{\theta})$. A sparse Gaussian process introduces inducing points, which includes inducing inputs $\boldsymbol{z} =$

$(z_1, z_2, \ldots, z_M)$ and inducing variables $\boldsymbol{u} = (u_1, u_2, \ldots, u_M)$ with relation such that $u_m = f(z_m)$ for $m = 1, 2, \ldots, M$.

We summarize a bunch of state-of-the-art inducing-point based approaches in a unified view. Those approaches include subset of regression (SoR) (Quiñonero-Candela and Rasmussen, 2005), deterministic training conditional approximation (DTC) (Seeger, 2003), fully independent training conditional approaximation (FITC) (Snelson and Ghahramani, 2006b), sparse Gaussian process regression (SGPR) (Titsias, 2009) and stochastic variational inference for sparse Gaussian process (SVGP) (Hensman et al., 2013). We denote the covariance matrix between inputs $\boldsymbol{u}$ and inputs $\boldsymbol{v}$ as $K_{u,v}$. We also denote $Q_{x,x} = K_{x,z}K_{x,x}^{-1}K_{z,x}$, $\tilde{K} = K_{x,x} - Q_{x,x}$ and $\boldsymbol{k}_i = (K_{x,z})_{i,:}$. As for the SVGP approach, the variational distribution of inducing variables is modeled as a Gaussian distribution $\boldsymbol{u} \sim \mathcal{N}(\boldsymbol{u}|\boldsymbol{m}, S)$. All sparse Gaussian process approaches attempt to maximize their objective function L_1 in the training procedure. Objective functions of different models are summarized in Table 3.4.

Table 3.4: The objective function in the training step for different models.

MODEL	L_1
SoR/DTC	$\log\left(\mathcal{N}(\boldsymbol{y}\|\boldsymbol{0}, Q_{\boldsymbol{x},\boldsymbol{x}} + \sigma^2\boldsymbol{I})\right)$
FITC	$\log\left(\mathcal{N}(\boldsymbol{y}\|\boldsymbol{0}, Q_{\boldsymbol{x},\boldsymbol{x}} + \mathrm{diag}(K_{\boldsymbol{x},\boldsymbol{x}} - Q_{\boldsymbol{x},\boldsymbol{x}}) + \sigma^2\boldsymbol{I})\right)$
SGPR	$\log\mathcal{N}(\boldsymbol{y}\|\boldsymbol{0}, Q_{\boldsymbol{x},\boldsymbol{x}} + \sigma^2\boldsymbol{I}) - \frac{1}{2\sigma^2}\mathrm{tr}(\tilde{K})$
SVGP	$\sum_{i=1}^{n}\left(\log\left(\mathcal{N}(y_i\|K_{\boldsymbol{x},\boldsymbol{z}}K_{\boldsymbol{z},\boldsymbol{z}}^{-1}\boldsymbol{m}, \sigma^2)\right) - \frac{1}{2\sigma^2}SK_{\boldsymbol{z},\boldsymbol{z}}^{-1}\boldsymbol{k}_i\boldsymbol{k}_i^T K_{\boldsymbol{z},\boldsymbol{z}}^{-1} - \frac{1}{2}\tilde{K}_{i,i}\right) - \mathrm{KL}(q(\boldsymbol{u})\|p(\boldsymbol{u}))$

Expressions of the predictive mean of those approaches can be written as a weighted sum of kernels centered at the inducing inputs. Mathematically, given a new input x^*, the predictive mean at x^* is

$$\mu^*(x^*) = \sum_{i=1}^{M} c_i K_{x^*,z_i} = K_{x^*,z}\boldsymbol{c}\,. \tag{3.77}$$

The weights $\boldsymbol{c}$ in (3.77) are summarized in Table 3.5. In a full Gaussian

process with zero mean, the predictive mean is $\mu^*(x^*) = K_{x^*,x}K_{x,x}^{-1}\boldsymbol{y}$, which is also a weighted sum of kernels. But it has N kernels centering at inputs $\boldsymbol{x}$. To get a better approximation of the full Gaussian process, it is necessary to enforce the space spanned by $K_{\cdot,z}$ to be similar to the space spanned by $K_{\cdot,x}$. Therefore, we prefer to minimize the difference of the distributions of the inducing inputs $\boldsymbol{z}$ and the inputs $\boldsymbol{x}$.

Table 3.5: Posterior mean of different sparse Gaussian process models.

methods	$\boldsymbol{c}$
SoR/DTC	$K_{z,z}^{-1}K_{z,x}(Q_{x,x} + \sigma^2 I)^{-1}\boldsymbol{y}$
FI(T)C	$K_{z,z}^{-1}K_{z,x}(Q_{x,x} + \mathrm{diag}(K_{x,x} - Q_{x,x}) + \sigma^2 I)^{-1}\boldsymbol{y}$
SGPR	$K_{z,z}^{-1}\sigma^{-2}(K_{z,z} + \sigma^{-2}K_{z,x}K_{x,z})^{-1}K_{z,x}\boldsymbol{y}$
SVGP	$K_{z,z}^{-1}\boldsymbol{m}$

Directly optimizing the inducing inputs in those models with objective functions in Table 3.4 is difficult, because of the non-convexity of those functions. Motivated by approximating the predictive mean from a kernel perspective, we propose a regularization framework, which is feasible for all methods above and contributes to a better approximation. In general, the regularization framework is as follows:

$$L_2 \triangleq L_1 + \lambda D(\boldsymbol{x}, \boldsymbol{z}) \tag{3.78}$$

where λ is a regularization weight and D is a measurement between the distributions of training inputs $\boldsymbol{x}$ and inducing inputs $\boldsymbol{z}$. We propose two approaches for D. The first approach is a non-parametric method, proposed as

$$D_1(\boldsymbol{x}, \boldsymbol{z}) = \min_s \sum_{n=1}^{N} \|x_n - z_{s(n)}\| \tag{3.79}$$

where s is an assignment function, which assigns the closest inducing input to each input. D_1 describes the total distance between inputs $\boldsymbol{x}$ and inducing inputs

$\boldsymbol{z}$ under the optimal assignment. The smaller D_1 is, the more similar the two distributions are.

The second approach is a parametric method, proposed as

$$D_2(\boldsymbol{x}, \boldsymbol{z}) = \mathrm{KL}(\hat{q}(\boldsymbol{x})||\hat{q}(\boldsymbol{z})) \tag{3.80}$$

where q is a similar type of distribution family. For computational convenience, we suppose q belongs to a Gaussian distribution. In such case, the estimate of q for data is summarized by its sample mean and sample variance and the Kullback-Leibler divergence measures the similarity between the two estimated distributions of inputs $\boldsymbol{x}$ and inducing inputs $\boldsymbol{z}$. One drawback of the parametric approach is that the variational distribution may not be flexible enough to model inputs or inducing inputs. But this method is more compelling in latent variable models, where the inputs are latent and are reasonably modeled using Gaussian distributions.

For both proposed methods, selecting an optimal regularization weight is important, because it balances the information from the model likelihood and the sampling information. The general method for selection of the regularization weight is through cross validation. In particular, we propose a regularization candidate pool $\{\lambda_i\}$ and select the optimal λ_i with the best prediction accuracy on the validation data. In general, the selection of λ depends on the choice of your measurement D. As for D_1, λ does not depend on data size N, while as for D_2, λ should be proportional to data size N. This is because in general L_1 is proportional to data size N, suggesting it is necessary to let λD be proportional to data size N.

We illustrate our regularization approach on 1-D synthetic data, where we uniformly generate 100 inputs $\boldsymbol{x}$ on the unit interval $[0, 1]$ and corresponding

observations are generated from the specified model

$$\begin{aligned} y|f &\sim \mathcal{N}(y|f, 0.1^2) \\ f &= \sin(2x) + 0.2\cos(22x)\,. \end{aligned}$$

We take another 100 evenly spaced inputs as testing inputs $\boldsymbol{x}_{test}$ on $[0, 1]$ with corresponding outputs $\boldsymbol{f}_{test}$ without noise as their ground-truth for testing. After we generate data, we set a Matern kernel with $\nu = \frac{3}{2}$ for $C(\boldsymbol{\theta})$ and set $M = 10$. Then we evaluate our regularization through three models.

The first framework M_1 fixes inducing inputs as evenly spaced inputs on $[0, 1]$ and optimizes hyper-parameters through maximizing L_1. The second framework M_2 optimizes both inducing points and hyper-parameters through maximizing L_2 and the last framework M_3 optimizes both inducing points and hyper-parameters through maximizing L_2 with regularization term D_1, and λ is selected by 5 fold cross-validation under the regularization candidate pool $[0, 1, \ldots, 9]$. The last model M_3 is the generalized version of M_1 and M_2, where M_1 fixes $\boldsymbol{z}$ and M_2 fixes $\lambda = 0$.

We evaluate our regularization approach under different frameworks using the mean square error of prediction on the testing data, shown in Table 3.6. We also compare them with a fully GP model. It illustrates that our regularization approach, M_3, consistently contributes to better prediction.

Model	M_1	M_2	M_3
SoR/DTC	0.052	0.051	**0.042**
FITC	**0.049**	0.083	0.056
SGPR	0.052	**0.046**	**0.046**
SVGP	0.051	**0.042**	**0.042**
fully GP	0.045		

Table 3.6: Predictive root mean square error under different models.

Moreover, we quantify the uncertainty of prediction using two measures. One is the coverage rate and the other is the average length of the 95% credible intervals. As for the coverage rate, the more close to 95% the coverage rate is, the better prediction performance it has. As for the average length, the small the value is, the better prediction performance it has. We summarize uncertainty prediction measures under the same three frameworks M_1, M_2 and M_3 for SoR, DTC, FITC, SGPR and SVGP methods in Table 3.7. We also compare it with a fully Gaussian process.

Model	Coverage rate			Average length of 95% credible intervals		
	M_1	M_2	M_3	M_1	M_2	M_3
SoR	0.81	0.75	0.86	0.131	0.106	0.129
DTC	0.97	0.94	0.99	0.342	1.001	0.359
FITC	0.98	0.97	0.96	0.268	0.501	0.346
SGPR	0.97	0.97	0.98	0.317	0.505	0.444
SVGP	0.94	0.92	0.99	0.211	0.211	0.213
fully GP	0.95			0.171		

Table 3.7: Coverage rate and average length of 95% credible intervals for three frameworks under different models

We next extend our regularization into latent variable models and focus on variational inference, and also show the equivalence between our regularized inference and a variational inference on a related hierarchical empirical Bayesian model. In the latent variable models, the model is not identifiable, our model would focus on the learning the manifold of data instead of uncertainty quantification.

3.3 Regularization for Latent Sparse Gaussian Processes

The Gaussian process latent variable model (GPLVM) is a powerful dimensionality reduction approach (Lawrence, 2004; Ek et al., 2007) and it is a base model for many sophisticated models (Lawrence and Moore, 2007; Urtasun and Darrell, 2007; Lawrence and Quiñonero Candela, 2006; Damianou et al., 2016). However, this model has two shortcomings. One is that the model is difficult to scale and the other is that the model fitting is sensitive to the initialization for both inducing inputs and embedding inputs. In practice, the principal component analysis (PCA) initialization for embedding inputs and K-means initialization for inducing inputs are standard procedures.

We extend the regularization of sparse Gaussian process into latent variable models to archive a better approximation by maintaining the similarity between the distribution of inducing inputs and embedding inputs while maximizing the marginal likelihood lower bound.

3.3.1 A Unified View of Sparse Latent Gaussian Processes

Suppose $Y \in \mathbb{R}^{N \times D}$ are observations with latent variables $F \in \mathbb{R}^{N \times D}$, where N is the number of observations and D is the dimension size. Let $X \in \mathbb{R}^{N \times Q}$ be latent variables for $\boldsymbol{F}$, where Q is the dimension size of the latent space. Assuming all features are conditional independent on $\boldsymbol{X}$, the GPLVM model is

$$\begin{aligned} y_{nd}|f_{nd} &\sim \mathcal{N}(y_{nd}|f_{nd}, \sigma^2 = \beta^{-1}) \,, \\ f_{nd} &= \mathcal{F}_d(\boldsymbol{x}_n) \,, \\ \mathcal{F}_d &\overset{iid}{\sim} \mathrm{GP}(0, C(\boldsymbol{\theta})) \text{ for } d = 1, 2, \ldots, D \,, \end{aligned} \tag{3.81}$$

with Gaussian priors for the latent variables X, $p(X) = \prod_{n=1}^{N} \mathcal{N}(\boldsymbol{x}_n|\boldsymbol{\mu}_X, \Sigma_X)$ where $\boldsymbol{x}_n$ is the n^{th} row of X. Specifically, we set $\boldsymbol{\mu}_X = \mathbf{0}$ and $\Sigma_X = I_Q$. Titsias (2009); Titsias and Lawrence (2010) propose a variational sparse GP formulation by introducing D separate sets of M inducing variables $U \in \mathbb{R}^{M\times D}$ evaluated at a set of inducing inputs $Z \in \mathbb{R}^{M\times Q}$. And Titsias and Lawrence (2010) and Hensman et al. (2013) propose the same tractable variational structure:

$$q(F, U, X) = \prod_{d=1}^{D} \left(p(\boldsymbol{f}_d|\boldsymbol{u}_d, X)q(\boldsymbol{u}_d)\right) q(X)\,, \tag{3.82}$$

where $\boldsymbol{f}_d$ is the d^{th} column of F and $\boldsymbol{u}_d$ is the d^{th} column of U. To take the advantage of conjugacy, all variational distributions are set as the same family as their prior distribution. Specifically, $q(X) = \prod_{n=1}^{N} \mathcal{N}(\boldsymbol{x}_n|\boldsymbol{\mu}_n, \Sigma_n)$. Then the evidence lower bound (ELBO) is

$$\text{ELBO} = \sum_{d=1}^{D} E_{q(F,U,X)} \log p(\boldsymbol{y}_d|\boldsymbol{f}_d) - \text{KL}(q(U)||p(U)) - \text{KL}(q(X)||p(X))\,.$$

Titsias and Lawrence (2010) derives the variational bound by marginalizing the optimal $q(U)$ based on the SGPR in Titsias (2009). After marginalization, the ELBO is derived as

$$\text{ELBO}_1 = \sum_{d=1}^{D} E_{q(X)}\Big(\log \mathcal{N}(\boldsymbol{y}_d|\mathbf{0}, K_{NM}K_{MM}^{-1}K_{MN} + \beta^{-1}I) - \frac{\beta}{2}\text{tr}(Q)\Big) - \text{KL}(q(X)||p(X))$$

where $Q = K_{NN} - K_{NM}K_{MM}^{-1}K_{MN}$. The optimal lower bound can also be derived by reverse Jensens' inequality in Titsias and Lawrence (2010).

On the other hand, by directly employing variational distributions $q(U) = \prod_{d=1}^{D} \mathcal{N}(\boldsymbol{u}_d|\boldsymbol{m}_d, S_d)$, we extend the univariate latent Gaussian process in Hensman et al. (2013) to multivariate latent Gaussian processes where data on different dimensions share the same embedding inputs. We denote the variational lower

bound as ELBO_2 with the expectation term derived as:

$$E_{q(F,U,X)} \log p(\boldsymbol{y}_d|\boldsymbol{f}_d) = E_{q(X)}\Big(\log \mathcal{N}(\boldsymbol{y}_d|K_{NM}K_{MM}^{-1}\boldsymbol{m}_d, \beta^{-1}I) - \frac{\beta}{2}\text{tr}(Q) - \frac{\beta}{2}\text{tr}(S_d\Lambda)\Big)$$

where $\Lambda = K_{MM}^{-1}K_{MN}K_{NM}K_{MM}^{-1}$. With sufficient statistics $\psi_0 = \text{tr}\langle K_{NN}\rangle_{q(X)}$, $\Psi_1 = \langle K_{NM}\rangle_{q(X)}$ and $\Psi_2 = \langle K_{MN}K_{NM}\rangle_{q(X)}$, the expectation term becomes tractable. Model fitting and model prediction are easy to derive, just as in Titsias and Lawrence (2010).

Comparing ELBO_1 with ELBO_2, ELBO_2 is scalable for large datasets via introducing the parametric distribution $q(U)$ and employing stochastic variational inference, while it is more difficult to optimize because more parameters are required to optimize.

However, for large and complicated datasets, variational inference may fail to capture the distribution of the embedding inputs via inducing inputs and result in poor model fitting and prediction. We will illustrate the benefits of regularization with different latent dimension settings and different ELBOs on two real datasets in the experiments section. To address this concern, we next propose an innovative regularization approach. We illustrate that inference using our modified lower bound is equivalent to inference on a lower bound in a related empirical Bayesian model under certain conditions.

3.3.2 Regularization in Latent Variable Models

With the same motivation as in the Sparse Gaussian process, we extend regularization into latent variable models by ensuring the inducing inputs capture the distribution of the embedding inputs. Therefore, we borrow the proposed regularizers to quantify the difference between the distribution of the inducing inputs and the distribution of the embedding inputs rather than the distribution

of deterministic inputs, and penalize the difference in the objective function.

Generally, we define an objective function called modified evidence lower bound as

$$\text{MELBO} = \text{ELBO} - \lambda R \tag{3.83}$$

where λ is a regularization weight and R is a regularization term which measures the difference between the distribution of the embedding inputs X and the distribution of the inducing inputs Z. As λ increases, the optimization emphasizes more similarity in the two distributions.

We choose the parametric regularizer D_2 in (3.80) for R rather than the non-parametric regularizer D_1 in (3.79), because both inducing inputs and embedding inputs are random and need to be optimized, so it is more practical to use a parametric distribution to approximate them. Given the cheap computation of the parametric approach, we choose it as the proposed regularizer in latent variable models.

Specifically, in the variational inference framework, to approximate the distribution of inducing inputs and embedding inputs, we build a global model for the variational mean of X such that every $\boldsymbol{\mu}_n$ has an independent identical Gaussian distribution $p_X(\boldsymbol{\mu}_n) = \mathcal{N}(\boldsymbol{\mu}_n|\boldsymbol{\mu}_{\boldsymbol{\mu}}, \Sigma_{\boldsymbol{\mu}})$, and build another global model for the inducing points Z such that every $\boldsymbol{z}_m$ has an independent identical distribution $p_Z(\boldsymbol{z}_m) = \mathcal{N}(\boldsymbol{z}_m|\boldsymbol{\mu}_Z, \Sigma_Z)$. Then given $\boldsymbol{\mu}$ and Z, we derive the maximum likelihood estimates $\hat{\boldsymbol{\mu}}_{\boldsymbol{\mu}}, \hat{\Sigma}_{\boldsymbol{\mu}}$, $\hat{\boldsymbol{\mu}}_Z$ and $\hat{\Sigma}_Z$ using the mean and covariance matrix of $\{\boldsymbol{\mu}_n\}$ and $\{\boldsymbol{z}_m\}$. Therefore, we have estimated the distribution $q_X = \mathcal{N}(\hat{\boldsymbol{\mu}}_{\boldsymbol{\mu}}, \hat{\Sigma}_{\boldsymbol{\mu}})$ to summarize the global distribution of the embedding inputs and $q_Z = \mathcal{N}(\hat{\boldsymbol{\mu}}_Z, \hat{\Sigma}_Z)$ to summarize the global distribution of the inducing inputs Z.

According to the definition of D_2 with respect to q_X and q_Z, we define

$$R = \mathrm{KL}(q_Z||q_X)\,. \tag{3.84}$$

In (3.83), λ can be chosen by cross validation or be set as the number of inducing points as a rule of thumb. In practice, because the log likelihood in a standard ELBO is proportional to the number of data points N while the regularization term R does not depend on N, it implies the optimal value of regularization should depend on the data size. On the other hand, if we choose ELBO = ELBO_2, stochastic variational inference is available by updating local parameters with respect to data indexes and global parameters in each batch. Specifically, the modified lower bound can be written as

$$\begin{aligned}\mathrm{MELBO} = \sum_{n=1}^{N}\Bigg[\sum_{d=1}^{D} E_{q(\boldsymbol{x}_n)}\log\mathcal{N}(y_{nd}|k_{nM}K_{MM}^{-1}\boldsymbol{m}_d,\beta^{-1}) \\ - E_{q(\boldsymbol{x}_n)}\frac{\beta}{2}Q_{nn} - E_{q(\boldsymbol{x}_n)}\frac{\beta}{2}\mathrm{tr}(S_d\Lambda_n) - \frac{\lambda}{N}\mathrm{KL}(q_Z||q_X)\Bigg]\end{aligned} \tag{3.85}$$

where $\Lambda_n = K_{MM}^{-1}k_{Mn}k_{nM}K_{MM}^{-1}$.

3.3.3 Regularization Theory

This section discusses the underlying relationship between regularization in a sparse GPLVM and a related empirical Bayesian model. First, we display the related empirical Bayesian model with a prior on its inducing inputs Z and derive its variational lower bound. Then we illustrate that when regularization weight is equal to the number of inducing points, $\lambda = M$, maximizing the MELBO is equivalent to maximizing the variational lower bound in the empirical Bayesian model under a mild condition.

The related empirical Bayesian model is extended from (3.81) and (3.82). We put an informative prior on the inducing inputs and propose a variational distribution on them as

$$\begin{aligned} \boldsymbol{z}_m &\sim \mathcal{N}(\boldsymbol{z}_m|\hat{\boldsymbol{\mu}}_{\boldsymbol{\mu}}, \hat{\Sigma}_{\boldsymbol{\mu}}) \\ q(\boldsymbol{z}_m) &= \mathcal{N}(\boldsymbol{z}_m|\boldsymbol{\nu}_m, \Upsilon_m) \end{aligned}$$

where $\hat{\boldsymbol{\mu}}_{\boldsymbol{\mu}}$, $\hat{\Sigma}_{\boldsymbol{\mu}}$ are estimates using sample mean and sample covariance matrix of $\{\boldsymbol{\mu}_n\}$.

The empirical Bayesian model is displayed using a graphical representation in Figure 3.4. The prior of inducing points borrow the information from the variational mean of embedding inputs $\boldsymbol{\mu}$.

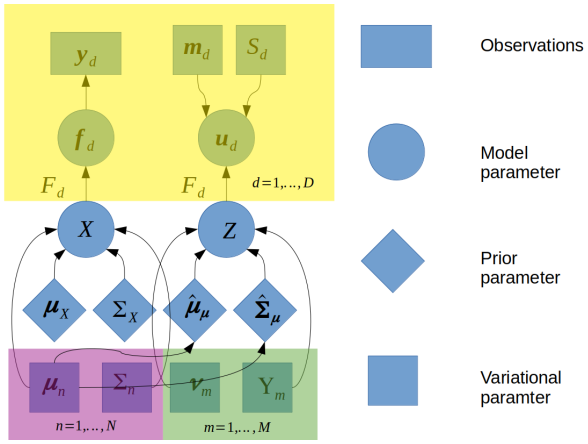

Figure 3.4: Graphical representation for the empirical Bayesian model.

The variational joint distribution is structurally decomposed and defined as $q(F, U, X, Z) = q(Z)q(X)q(U)p(F|Z, X, U)$. Then the variational lower bound is derived as

$$\begin{aligned} \log p(Y) &\geq \mathrm{E}_{q(F,U,X,Z)} \log p(Y|F) - \mathrm{KL}(q(Z)||p(Z)) - \mathrm{KL}(q(X)||p(X)) - \mathrm{KL}(q(U)||p(U)) \\ &\triangleq \mathrm{ELBO}_{\mathrm{EB}} \end{aligned} \tag{3.86}$$

We define $\hat{\boldsymbol{\mu}}_{\boldsymbol{\nu}}$ and $\hat{\Sigma}_{\boldsymbol{\nu}}$ as sample mean and sample covariance matrix of $\{\boldsymbol{\nu}_m\}$ and define a distribution family for $q(Z)$ such that $\Upsilon_m = \epsilon I$ for $m = 1, \ldots, M$. We justify that under the assumption that there exist $K > 0$ such that $|\hat{\Sigma}_{\boldsymbol{\nu}}| < K$, we have following three lemmas and one theorem.

Lemma 1. *Assume* $q(\boldsymbol{z}_m) = \mathcal{N}(\boldsymbol{\nu}_m, \epsilon I)$. *As* $\epsilon \to 0$, $\boldsymbol{z}_m \xrightarrow{p} \boldsymbol{\nu}_m$.

Proof. Since $\forall \epsilon_0 > 0$,

$$\begin{aligned}
\lim_{\epsilon \to 0} p(|\boldsymbol{z}_m - \boldsymbol{\nu}_m| > \epsilon_0) &= \lim_{\epsilon \to 0} p(|\frac{\boldsymbol{z}_m - \boldsymbol{\nu}_m}{\epsilon}| > \frac{\epsilon_0}{\epsilon}) \\
&= 2 \lim_{\epsilon \to 0} (1 - \Phi(\frac{\epsilon_0}{\epsilon}))^Q \\
&= 0\,,
\end{aligned}$$

we conclude that $\boldsymbol{z}_m \xrightarrow{p} \boldsymbol{\nu}_m$. □

Lemma 2. *In the variational lower bound (3.86),*

$$\mathrm{KL}(q(Z)||p(Z)) \leq A - B - C \tag{3.87}$$

where

$$\begin{aligned}
A &= \frac{M}{2}(\log|\hat{\Sigma}_{\boldsymbol{\mu}}| + \log|\hat{\boldsymbol{\Sigma}}_{\boldsymbol{\nu}}| + Q) \\
&\quad + \frac{1}{2}\left(\sum_{m=1}^{M} (\boldsymbol{\nu}_m - \hat{\boldsymbol{\mu}}_{\boldsymbol{\mu}})^T \hat{\Sigma}_{\boldsymbol{\mu}}^{-1} (\boldsymbol{\nu}_m - \hat{\boldsymbol{\mu}}_{\boldsymbol{\mu}})\right), \\
B &= \frac{M}{2}(Q \log \epsilon - \log K), \\
C &= \frac{2\epsilon}{M \mathrm{tr}(\hat{\Sigma}_{\boldsymbol{\mu}}^{-1})}\,.
\end{aligned}$$

Proof.

$$\begin{aligned}\mathrm{KL}(q(\boldsymbol{Z})||p(\boldsymbol{Z})) &= A - \frac{M}{2}(Q\log\epsilon - \log|\hat{\boldsymbol{\Sigma}}_{\boldsymbol{\nu}}|) - C \\ &\leq A - B - C\end{aligned}$$

because of the finite variance assumption that $|\hat{\boldsymbol{\Sigma}}_{\boldsymbol{\nu}}| < K$. □

Lemma 3.

$$M\mathrm{KL}(q_Z||q_X) = \frac{M}{2}(\log|\hat{\Sigma}_{\boldsymbol{\mu}}| + \log|\hat{\boldsymbol{\Sigma}}_{\boldsymbol{z}}| + Q) + \frac{1}{2}\left(\sum_{m=1}^{M}(\boldsymbol{z}_m - \hat{\boldsymbol{\mu}}_{\boldsymbol{\mu}})^T\hat{\Sigma}_{\boldsymbol{\mu}}^{-1}(\boldsymbol{z}_m - \hat{\boldsymbol{\mu}}_{\boldsymbol{\mu}})\right).$$

Proof.

$$\begin{aligned}
\mathrm{KL}(q_Z||q_X) &= \frac{1}{2}\left[\log\frac{|\hat{\Sigma}_{\boldsymbol{\mu}}|}{|\hat{\Sigma}_Z|} - Q + \mathrm{tr}(\hat{\Sigma}_{\boldsymbol{\mu}}^{-1}\hat{\Sigma}_Z) + (\hat{\boldsymbol{\mu}}_{\boldsymbol{\mu}} - \hat{\boldsymbol{\mu}}_Z)^T\hat{\Sigma}_{\boldsymbol{\mu}}^{-1}(\hat{\boldsymbol{\mu}}_{\boldsymbol{\mu}} - \hat{\boldsymbol{\mu}}_Z)\right] \\
&= \frac{1}{2}\left[\log\frac{|\hat{\Sigma}_{\boldsymbol{\mu}}|}{|\hat{\Sigma}_Z|} - Q + \mathrm{tr}\left(\hat{\Sigma}_{\boldsymbol{\mu}}^{-1}((\hat{\boldsymbol{\mu}}_{\boldsymbol{\mu}} - \hat{\boldsymbol{\mu}}_Z)(\hat{\boldsymbol{\mu}}_{\boldsymbol{\mu}} - \hat{\boldsymbol{\mu}}_Z)^T + \hat{\Sigma}_Z)\right)\right] \\
&= \frac{1}{2}\left[\log\frac{|\hat{\Sigma}_{\boldsymbol{\mu}}|}{|\hat{\Sigma}_Z|} - Q + \frac{1}{M}\mathrm{tr}\left(\hat{\Sigma}_{\boldsymbol{\mu}}^{-1}(M(\hat{\boldsymbol{\mu}}_{\boldsymbol{\mu}} - \hat{\boldsymbol{\mu}}_Z)(\hat{\boldsymbol{\mu}}_{\boldsymbol{\mu}} - \hat{\boldsymbol{\mu}}_Z)^T + \sum_{m=1}^{M}(\boldsymbol{u}_m - \hat{\boldsymbol{\mu}}_Z)(\boldsymbol{u}_m - \hat{\boldsymbol{\mu}}_Z)^T\right)\right] \\
&= \frac{1}{2}\Bigg[\log\frac{|\hat{\Sigma}_{\boldsymbol{\mu}}|}{|\hat{\Sigma}_Z|} - Q + \frac{1}{M}\mathrm{tr}\Big(\hat{\Sigma}_{\boldsymbol{\mu}}^{-1}(M\hat{\boldsymbol{\mu}}_{\boldsymbol{\mu}}\hat{\boldsymbol{\mu}}_{\boldsymbol{\mu}}^T - M\hat{\boldsymbol{\mu}}_{\boldsymbol{\mu}}\hat{\boldsymbol{\mu}}_Z^T - M\hat{\boldsymbol{\mu}}_Z\hat{\boldsymbol{\mu}}_{\boldsymbol{\mu}}^T + M\hat{\boldsymbol{\mu}}_Z\hat{\boldsymbol{\mu}}_Z^T \\
&\quad + \sum_{m=1}^{M}\boldsymbol{u}_m\boldsymbol{u}_m^T - (\sum_{m=1}^{M}\boldsymbol{z}_m)\hat{\boldsymbol{\mu}}_Z^T - \hat{\boldsymbol{\mu}}_Z(\sum_{m=1}^{M}\boldsymbol{z}_m)^T + M\hat{\boldsymbol{\mu}}_Z\hat{\boldsymbol{\mu}}_Z^T\Big)\Bigg] \\
&= \frac{1}{2}\left[\log\frac{|\hat{\Sigma}_{\boldsymbol{\mu}}|}{|\hat{\Sigma}_Z|} - Q + \frac{1}{M}\mathrm{tr}\left(\hat{\Sigma}_{\boldsymbol{\mu}}^{-1}(\sum_{m=1}^{M}\boldsymbol{z}_m\boldsymbol{z}_m^T - M\hat{\boldsymbol{\mu}}_{\boldsymbol{\mu}}\hat{\boldsymbol{\mu}}_Z^T - M\hat{\boldsymbol{\mu}}_Z\hat{\boldsymbol{\mu}}_{\boldsymbol{\mu}}^T + M\hat{\boldsymbol{\mu}}_X\hat{\boldsymbol{\mu}}_{\boldsymbol{\mu}}^T)\right)\right] \\
&= \frac{1}{2}\left[\log\frac{|\hat{\Sigma}_{\boldsymbol{\mu}}|}{|\hat{\Sigma}_Z|} - Q + \frac{1}{M}\mathrm{tr}\left(\hat{\Sigma}_{\boldsymbol{\mu}}^{-1}(\sum_{m=1}^{M}\boldsymbol{z}_m\boldsymbol{z}_m^T - \hat{\boldsymbol{\mu}}_{\boldsymbol{\mu}}(\sum_{m=1}^{M}\boldsymbol{z}_m)^T - (\sum_{m=1}^{M}\boldsymbol{z}_m)\hat{\boldsymbol{\mu}}_{\boldsymbol{\mu}}^T + M\hat{\boldsymbol{\mu}}_{\boldsymbol{\mu}}\hat{\boldsymbol{\mu}}_{\boldsymbol{\mu}}^T)\right)\right] \\
&= \frac{1}{2}\left[\log\frac{|\hat{\Sigma}_{\boldsymbol{\mu}}|}{|\hat{\Sigma}_Z|} - Q + \frac{1}{M}\mathrm{tr}\left(\hat{\Sigma}_{\boldsymbol{\mu}}^{-1}(\sum_{m=1}^{M}\boldsymbol{z}_m\boldsymbol{z}_m^T - \hat{\boldsymbol{\mu}}_{\boldsymbol{\mu}}(\sum_{m=1}^{M}\boldsymbol{u}_m)^T - (\sum_{m=1}^{M}\boldsymbol{z}_m)\hat{\boldsymbol{\mu}}_{\boldsymbol{\mu}}^T + M\hat{\boldsymbol{\mu}}_{\boldsymbol{\mu}}\hat{\boldsymbol{\mu}}_{\boldsymbol{\mu}}^T)\right)\right] \\
&= \frac{1}{2}\left[\log\frac{|\hat{\Sigma}_{\boldsymbol{\mu}}|}{|\hat{\Sigma}_Z|} - Q + \frac{1}{M}\mathrm{tr}\left(\hat{\Sigma}_{\boldsymbol{\mu}}^{-1}(\sum_{m=1}^{M}(\boldsymbol{z}_m - \hat{\boldsymbol{\mu}}_{\boldsymbol{\mu}})(\boldsymbol{z}_m - \hat{\boldsymbol{\mu}}_{\boldsymbol{\mu}})^T)\right)\right] \\
&= \frac{1}{M}\sum_{m=1}^{M}\frac{1}{2}\left[\log\frac{|\hat{\Sigma}_{\boldsymbol{\mu}}|}{|\hat{\Sigma}_Z|} - Q + (\boldsymbol{z}_m - \hat{\boldsymbol{\mu}}_{\boldsymbol{\mu}})^T\hat{\Sigma}_{\boldsymbol{\mu}}^{-1}(\boldsymbol{z}_m - \hat{\boldsymbol{\mu}}_{\boldsymbol{\mu}})\right].
\end{aligned}$$

Therefore,

$$
\begin{aligned}
M\mathrm{KL}(q_Z||q_X) &= \frac{1}{2}\sum_{m=1}^{M}\left[\log\frac{|\hat{\Sigma}_{\boldsymbol{\mu}}|}{|\hat{\Sigma}_Z|} - Q + (\boldsymbol{z}_m - \hat{\boldsymbol{\mu}}_{\boldsymbol{\mu}})^T\hat{\Sigma}_{\boldsymbol{\mu}}^{-1}(\boldsymbol{z}_m - \hat{\boldsymbol{\mu}}_{\boldsymbol{\mu}})\right] \\
&= \frac{M}{2}(\log|\hat{\Sigma}_{\boldsymbol{\mu}}| + \log|\hat{\boldsymbol{\Sigma}}_{\boldsymbol{z}}| + Q) + \frac{1}{2}\left(\sum_{m=1}^{M}(\boldsymbol{z}_m - \hat{\boldsymbol{\mu}}_{\boldsymbol{\mu}})^T\hat{\Sigma}_{\boldsymbol{\mu}}^{-1}(\boldsymbol{z}_m - \hat{\boldsymbol{\mu}}_{\boldsymbol{\mu}})\right).
\end{aligned}
$$

□

Theorem 3.3.1. Given Lemma 1, Lemma 2 and Lemma 3, as $\epsilon \to 0$, maximizing the variational lower bound in empirical Bayesian model is equivalent to maximizing the MELBO in the sparse GPLVM with respect to $Z, q(X)$ and $q(U)$.

Proof. In the empirical Bayesian model, denote all parameters as $\Theta = [\boldsymbol{\mu}, \boldsymbol{\Sigma}, \boldsymbol{m}, \boldsymbol{s}, \boldsymbol{\nu}, \boldsymbol{h}]$. $\boldsymbol{h}$ denote all hyper-parameters in GP kernels.

Because of Lemma 1,

$$
\lim_{\epsilon\to 0} E_{q(F,U,X,Z)}\log p(Y|F) = E_{q(F,U,X)}\log p(Y|F, Z = \boldsymbol{\nu}).
$$

And $\lim_{\epsilon\to 0} C = \frac{2}{M\mathrm{tr}(\hat{\Sigma}_{\mu}^{-1})}\lim_{\epsilon\to 0}\epsilon = 0$. Because of Lemma 2, we have a loose lower bound such that $\mathrm{ELBO}_{\mathrm{EB}} \geq \mathrm{ELBO}_{\mathrm{EB}} + \mathrm{KL}(q(\boldsymbol{Z})||p(\boldsymbol{Z})) - A + B + C \triangleq \mathrm{LELBO}_{\mathrm{EB}}$. Instead of directly maximizing $\mathrm{ELBO}_{\mathrm{EB}}$, we are maximizing the loose low bound $\mathrm{LELBO}_{\mathrm{EB}}$ and the optimal estimates are

$$
\begin{aligned}
\hat{\Theta} &= \arg\max_{\Theta}\lim_{\epsilon\to 0}\mathrm{LELBO}_{\mathrm{EB}} \\
&= \arg\max_{\Theta}\lim_{\epsilon\to 0} E_{q(F,U,X,Z)}\log p(Y|F) - \mathrm{KL}(q(X)||p(X)) - \mathrm{KL}(q(U)||p(U)) - A + B + C \\
&= \arg\max_{\Theta}\lim_{\epsilon\to 0} E_{q(F,U,X,Z)}\log p(Y|F) - \mathrm{KL}(q(X)||p(X)) - \mathrm{KL}(q(U)||p(U)) - A \\
&= \arg\max_{\Theta} E_{q(F,U,X)}\log p(Y|F, Z = \boldsymbol{\nu}) - \mathrm{KL}(q(X)||p(X)) - \mathrm{KL}(q(U)||p(U)) - A
\end{aligned}
$$

Due to Lemma 3, this optimization is equivalent to maximizing $\mathrm{ELBO} - M\mathrm{KL}(q_Z||q_X)$ which is exactly the MELBO defined in (3.83). Finally, due to Lemma 1, the $q(Z)$ in empirical Bayesian model converges to the same optimized Z as in the regularized sparse GPLVM. □

3.4 Experiments

We illustrate our regularization framework on three datasets. First, we show that regularization is necessary for sparse GP in latent variable models for a moderate dataset. Taking the Anuran Calls dataset for instance, we explore the regularization for two different lower bounds and also explore the regularization approach for different latent dimension sizes. Second, we illustrate the regularization approach for a large dataset with different numbers of inducing points, using the Flight dataset. Finally, we take the Driver Face dataset as an example of an application for high dimensional datasets. All optimizations employ the Limited-memory BFGS approach with maximum iteration number 1000.

3.4.1 Anuran Calls Example

We show that regularization improves inference on the Anuran Call dataset. This dataset is available from the UCI repository at `https://archive.ics.uci.edu/ml/datasets/Anuran+Calls+(MFCCs)`, where there are 7195 instances, and each instance has 22 attributes. We model all instances using a sparse latent Gaussian process and perform inference with and without regularization.

Specifically, we set the latent dimension size $Q = 5$ and use $M = 20$ inducing points in the multivariate latent Gaussian process model. We choose independent standard Gaussian distributions as the prior distributions of the inducing points. We employ the PCA approach for initialization of the embedding inputs and employ the K-means algorithm for initialization of the inducing inputs.

Regularization with ELBO_1

This section considers the optimal variational distribution of inducing variables, exploring three models with respect to the inducing inputs. The first model

is to fix the inducing inputs as the initial K-means' centroids. The second model is to treat the inducing inputs as trainable parameters in optimization. And the last model is to consider our proposed regularization approach, where λ is selected in $[1, 10, 100, 1000]$ through 5-fold cross-validation. After model training, we estimate embedding inputs as their variational mean $\hat{X} = \hat{\boldsymbol{\mu}}$ and reconstruct all observations by their mean given the estimated embedding inputs. Then we compare the root mean square errors (RMSE) for the fitting results. We also compare the similarity of distributions between embedding inputs X and inducing inputs Z by introducing averaged symmetric KL divergence criteria (ASKL). It is defined as $ASKL = \frac{1}{Q}\sum_{q=1}^{Q}(0.5\text{KL}(\hat{p}(\hat{X}_q)|\hat{p}(\hat{Z}_q) + 0.5\text{KL}(\hat{p}(\hat{Z}_q)|\hat{p}(\hat{X}_q))$, where $\hat{p}(X)$ is a Gaussian distribution fitted by X. Both RMSE and ASKL are summarized in Table 3.8 and the empirical distributions of estimated embedding inputs and inducing inputs for each dimension are shown in Figure 3.5. It demonstrates that our regularization approach is significantly better on both model fitting and latent input deployment.

	M_1	M_2	M_3
RMSE	0.0575	0.0438	**0.0434**
ASKL	2.5330	0.4213	**0.0111**

Table 3.8: Root mean square errors (RMSE) and averaged symmetric KL divergence (ASKL) for three different models with respect to inducing inputs under ELBO_1 setting. (Anuran Calls Example)

Regularization with ELBO_2

This section considers the parameterized variational distribution of inducing variables. Using the same model evaluation rules in the last section, we show RMSEs and ASKLs in Table 3.9 and the empirical distributions of estimated embedding inputs and inducing inputs for each dimension are shown in Figure 3.6.

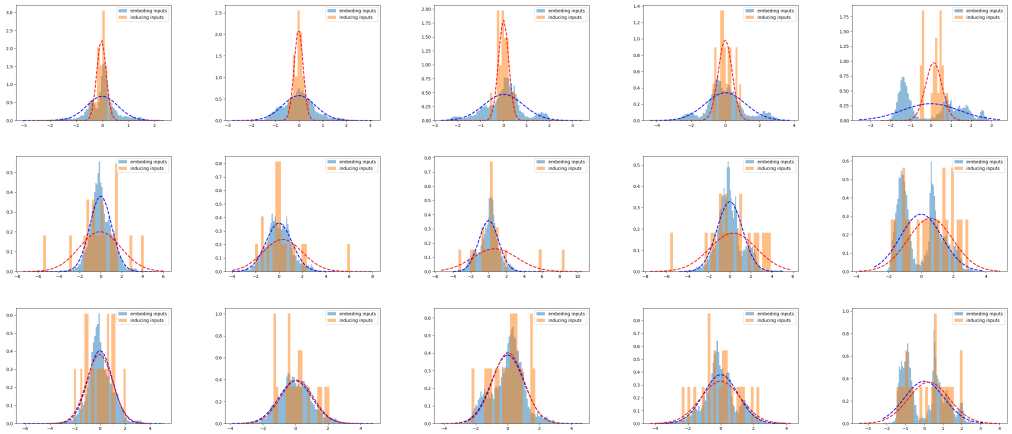

Figure 3.5: Empirical distributions of estimated embedding inputs and inducing inputs under ELBO_2 setting. Models 1 to 3 are shown by row and latent dimension 1 to Q are shown by column. (Anuran Calls Example)

It is obvious that our regularization approach achieves the best model fitting and the best latent input deployment. Also, it is clear to see that Model 2 using ELBO_2 has a significantly larger ASKL compared with using ELBO_1. This is because without marginalization, the non-convex objective function ELBO_2 involves more parameters and thus it is more difficult to optimize. However, with our proposed regularization, this model gets a comparable model fitting result with respect to that under the ELBO_1 setting.

	M_1	M_2	M_3
RMSE	0.0690	0.0521	**0.0453**
ASKL	2.6125	31.9826	**0.0766**

Table 3.9: Root mean square errors (RMSE) and averaged symmetric KL divergence (ASKL) for three different models with respect to inducing inputs under ELBO_2 setting. (Anuran Calls Example)

Regularization with different latent dimension sizes

We explore the benefits of regularization with respect to different latent dimension sizes under ELBO_2. Specifically, because of output dimension size $D =$

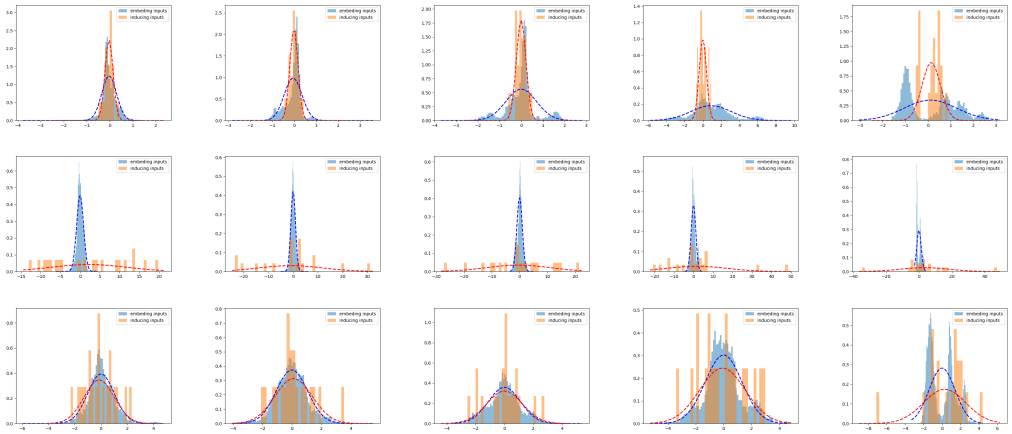

Figure 3.6: Empirical distributions of estimated embedding inputs and inducing inputs under ELBO_2 setting. Model 1 to 3 are shown by row and latent dimension 1 to Q are shown by column. (Anuran Calls Example)

22, we consider $Q = 2, 5, 10$ and set $\lambda = 1000$. The RMSEs and ASKLs are displayed in Table 3.10. The relative ratio of RMSEs, defined by (RMSE(N)-RMSE(R))/RMSE(N) as model improvement statistics, for $Q = 2, 5, 10$ are 18.7%, 7.1% and 9.9%. It shows that regularization is always contributing to better model fitting, especially when the latent dimension size is significantly smaller then the output dimension size.

	Q = 2	Q = 5	Q = 10
RMSE(N)	0.0851	0.0492	0.0354
RMSE(R)	**0.0692**	**0.0457**	**0.0319**
ASKL(N)	1022.5650	38.8489	162.4201
ASKL(R)	**1.2367**	**0.1403**	**0.0147**

Table 3.10: Root mean square errors (RMSE) and averaged symmetric KL divergence (ASKL) for model with regularization (R) and without regularization (N) under different latent dimension sizes $Q = 2, 5, 10$. (Anuran Calls Example)

3.4.2 Flight Example

We illustrate regularization for large datasets using the Flight data, which consists of every commercial flight in the USA from January to April 2008, information on 2 million flights. We choose to include into our model the same 8 variables as in Hensman et al. (2013). Instead of predicting the delay time using the 8 features, we focus on reconstructing noisy features. Specifically, we randomly choose 70k flights for training and another 10k flights to add noise for testing. In detail, we normalize all data with respect to each feature and randomly choose one feature to add white noise for each flight in those 10k flights. Our task is to reconstruct the features for those 10k flights and compare the reconstructed features with true features.

We choose the baseline model with variational lower bound ELBO_2 and set $\lambda = 1000$ for the regularization approach. We use batch gradient descent for inference. Root mean square errors for both 70k training data and 10k reconstruction data are displayed in Table 3.11. It illustrates that our regularization approach performs better for both model fitting and noisy data reconstruction. We also report the training time for both models. Our baseline model and regularization approach have the same time complexity $O(M^3)$. As M increases, the training time should have cubic growth if the number of training iterations is the same. In practice, for large datasets, we can use stochastic variational inference but that is beyond the scope of this work.

3.4.3 Driver Face Example

This section illustrates regularized sparse latent Gaussian processes for high dimensional data such as image data. We show our regularization framework on a Driver Face dataset, which is available from the UCI repository at `https://`

	M = 10	M = 20	M = 50
RMSE (TB)	0.77	0.66	0.66
RMSE (TR)	**0.65**	**0.64**	**0.61**
RMSE (RB)	0.82	0.67	0.68
RMSE (RR)	**0.67**	**0.66**	**0.63**
T (B)	2 min	15 min	55 min
T (R)	2 min	14 min	72 min

Table 3.11: Root mean square errors (RMSE) of training data/reconstruction data (T/R) for baseline model/regularized model (B/R). Training time (T) are available for both models. (Flight Example)

`archive.ics.uci.edu/ml/datasets/DrivFace`. It includes 606 samples of 80×80 pixels each, acquired over different days from 4 drivers (2 women and 2 men) with several facial features like glasses and beard. Each individual has around 150 images. Each pixel's value is in the unit interval $[0, 1]$. We use 2×2 max-pooling to reduce the original image size 80×80 to 40×40 as pre-processing.

Model Fitting

We employ a sparse latent Gaussian process with $ELBO_2$ as a baseline model and set latent dimension size $Q = 5$. We consider different inducing point sizes $M = 10, 20, 50, 100$ and different regularization weights $\lambda = 10, 100, 1000, 10000$. To compare model fitting with and without regularization, we employ RMSEs for the model evaluation and ASKLs to show the balance between inducing inputs and embedding inputs. Results are shown in Figure 3.7. As λ increases, the inducing inputs capture embedding points better. More importantly, with proper regularization, our model fitting results are always better than the baseline model for all cases of M.

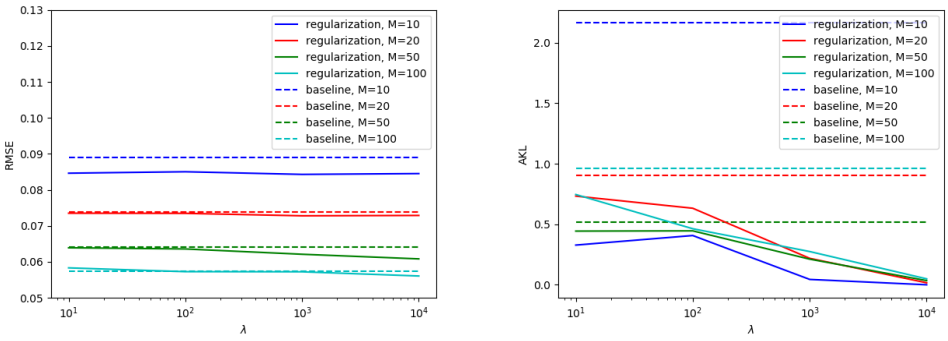

Figure 3.7: Root mean square errors and averaged symmetric KL divergence for the Driver Face dataset with different number of inducing points $M = 10, 20, 50, 100$ and with different regularization weights $\lambda = 10, 100, 1000, 10000$. Baseline model results are also provided. (Driver Face Example)

Image Denoising

In the section, we randomly select N images denoted as $\{\boldsymbol{y}_i\}_{i=1}^N$ and add noises on them denoted as $\{\tilde{\boldsymbol{y}}_i\}_{i=1}^N$. We also denote other images as $\{\boldsymbol{z}_i\}_{i=1}^M$. Assuming we do not know which images are blurred, we train both images with noise $\{\tilde{\boldsymbol{y}}_i\}_{i=1}^N$ and other images $\{\boldsymbol{z}_i\}_{i=1}^M$ together in our regularized latent sparse Gaussian process model unsupervisedly to obtain trained model as well as their corresponding latent variables on the low dimensional manifold. As images are projected to a low dimensional manifold, the key features would be kept while the noise features would be discarded. Then based on the estimated latent variables of the images with noise, we reconstruct their images through back transferring the corresponding latent variables via our trained regularized latent Gaussian process model.

We apply our regularized latent sparse Gaussian process for the image denoising task in this section. First, we randomly select six images, in which we randomly select 50 pixels to add white noise and clip them into a unit interval $[0, 1]$. Six images of them with and without noise are displayed in Figure 3.8.

We train the whole dataset under different settings with respect to the number

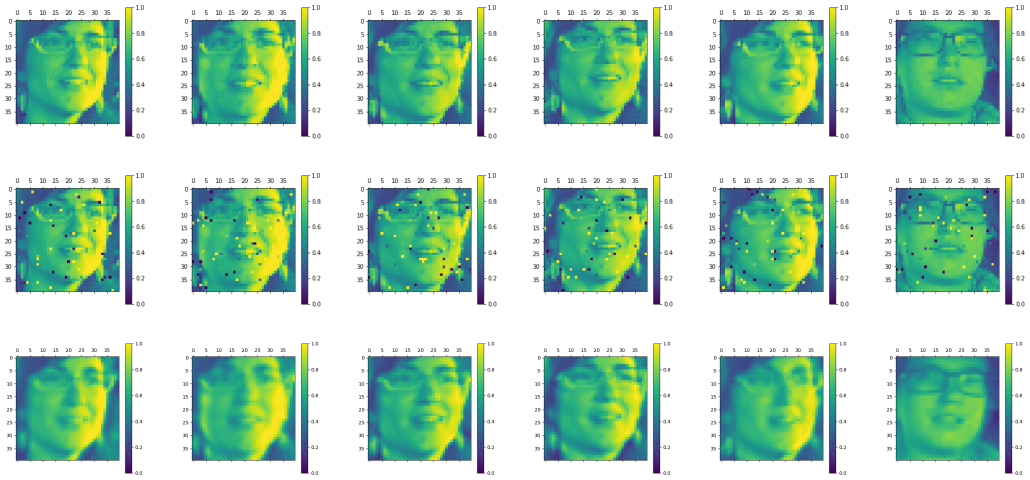

Figure 3.8: Six real images are shown in the first row. Corresponding noisy images and reconstructed images are displayed in the second row and the third row separately. (Driver Face Example)

of inducing points $M = 10, 20, 50, 100$. The regularization weight is optimally selected in the set $(10, 100, 1000, 10000)$. After our model is trained, we reconstruct the six noisy images and compare them with their true images. The RMSEs for the six noisy images are displayed in Table 3.12. It shows that our regularized model performs better than the corresponding baseline model for model prediction in all cases of $M = 10, 20, 50, 100$. We displayed the best results from our models in Figure 3.8.

M	10	20	50	100
Baseline model	0.1221	0.1085	0.1033	0.0982
Regularized model	**0.1195**	**0.1079**	**0.1024**	**0.0976**

Table 3.12: Root mean square errors for the six noisy images under different number of inducing points Ms. (Driver Face Example)

We also considered the image denoising task on more blurred figures. Specifically, we randomly select six images and randomly select a 20 by 20 block region to add which noise and clip them in a unit interval. Then we consider the different setting of our model $M = 10, 20$ and regularization weights $\lambda =$

$10, 100, 1000, 10000$. We displayed the best results from our models as well as both real and noised figures in Figure 3.9.

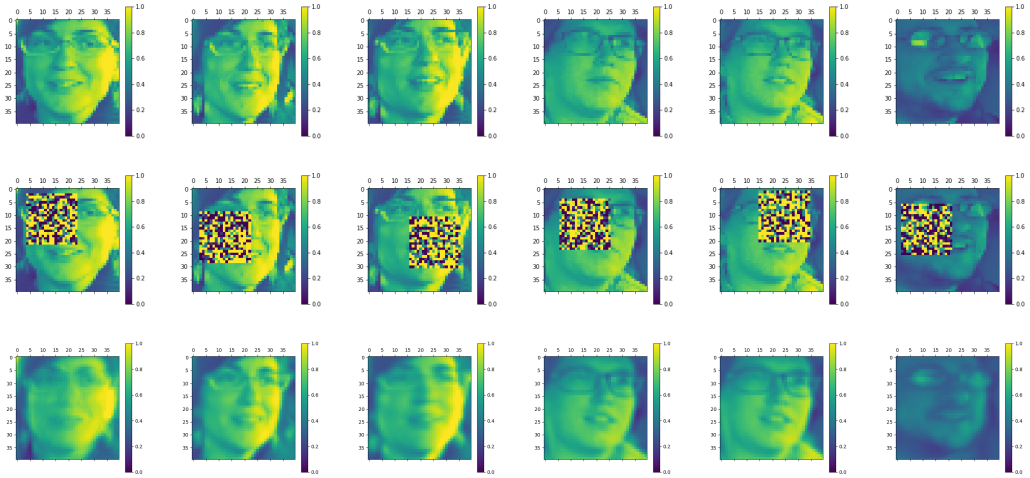

Figure 3.9: Six real images are shown in the first row. Corresponding noisy images and reconstructed images are displayed in the second row and the third row separately. (Driver Face Example)

Moreover, we are interested in how is the model performance under different levels of noises. In detail, we randomly select six images and add six different levels of Gaussian noises with scale $0.01, 0.02, 0.05, 0.1, 0.2$ and 0.5. Then we clip them in a unit interval. We consider the number of inducing points $M = 10$ and regularization weights $\lambda = 10, 100, 1000, 10000$. We select the best reconstruction results shown in Figure 3.10 and Figure 3.11. To compare them with original and training images, we also shown them in the same figure.

Figure 3.10 and Figure 3.11 show that the regularized latent Gaussian processes model is robust to noises while for the image with little noise, the reconstruction may make it more blurred.

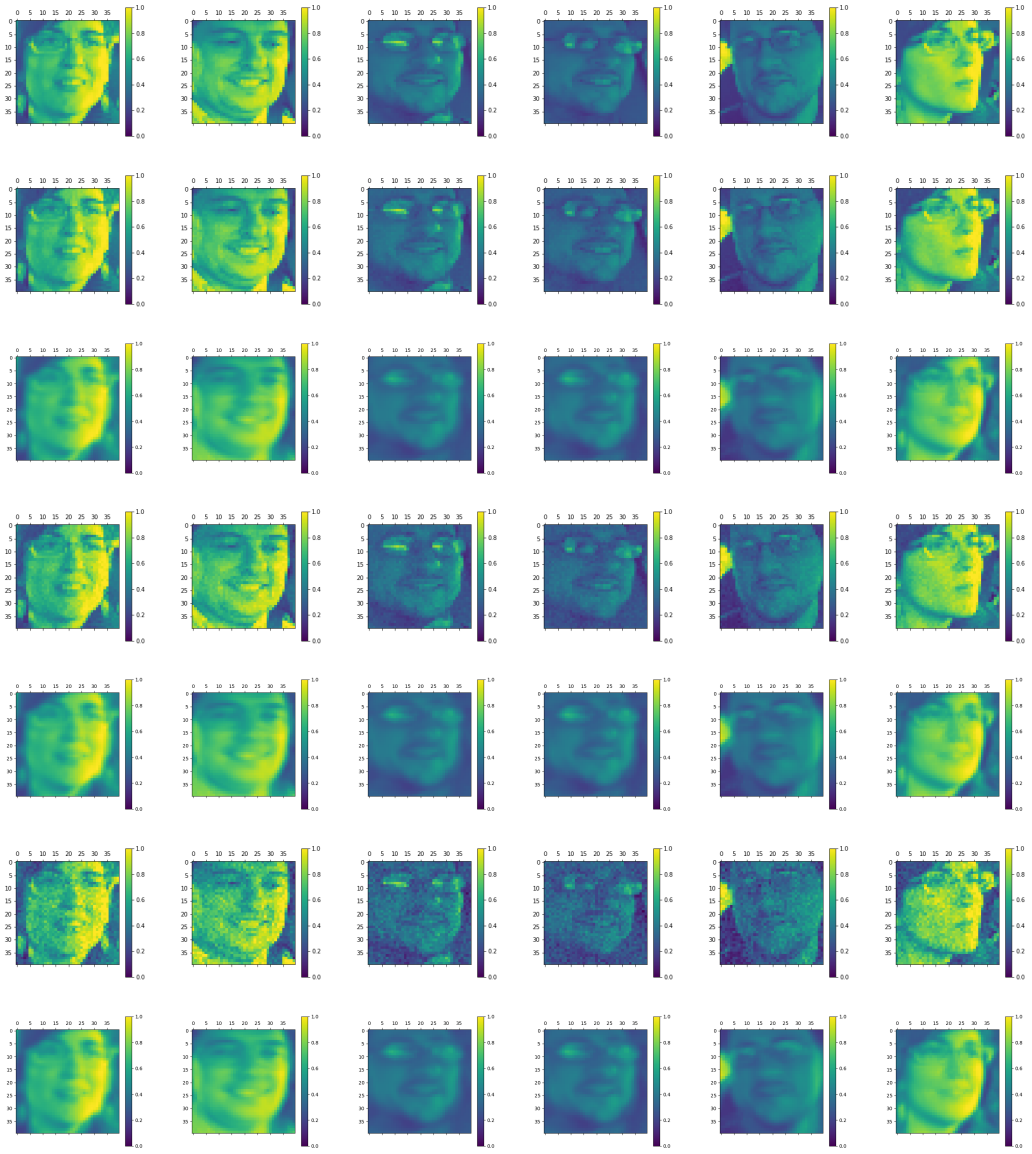

Figure 3.10: Six real images are shown in the first row. Corresponding noisy images and reconstructed images are displayed in the next two rows iteratively for three different scales $\sigma = 0.01, 0.02$ and 0.05. (Driver Face Example)

3.5 Conclusion

Regularization is necessary for sparse Gaussian processes especially in latent variable models. Our regularization approach improves global optimization in model fitting and achieves better model prediction. In the case of latent vari-

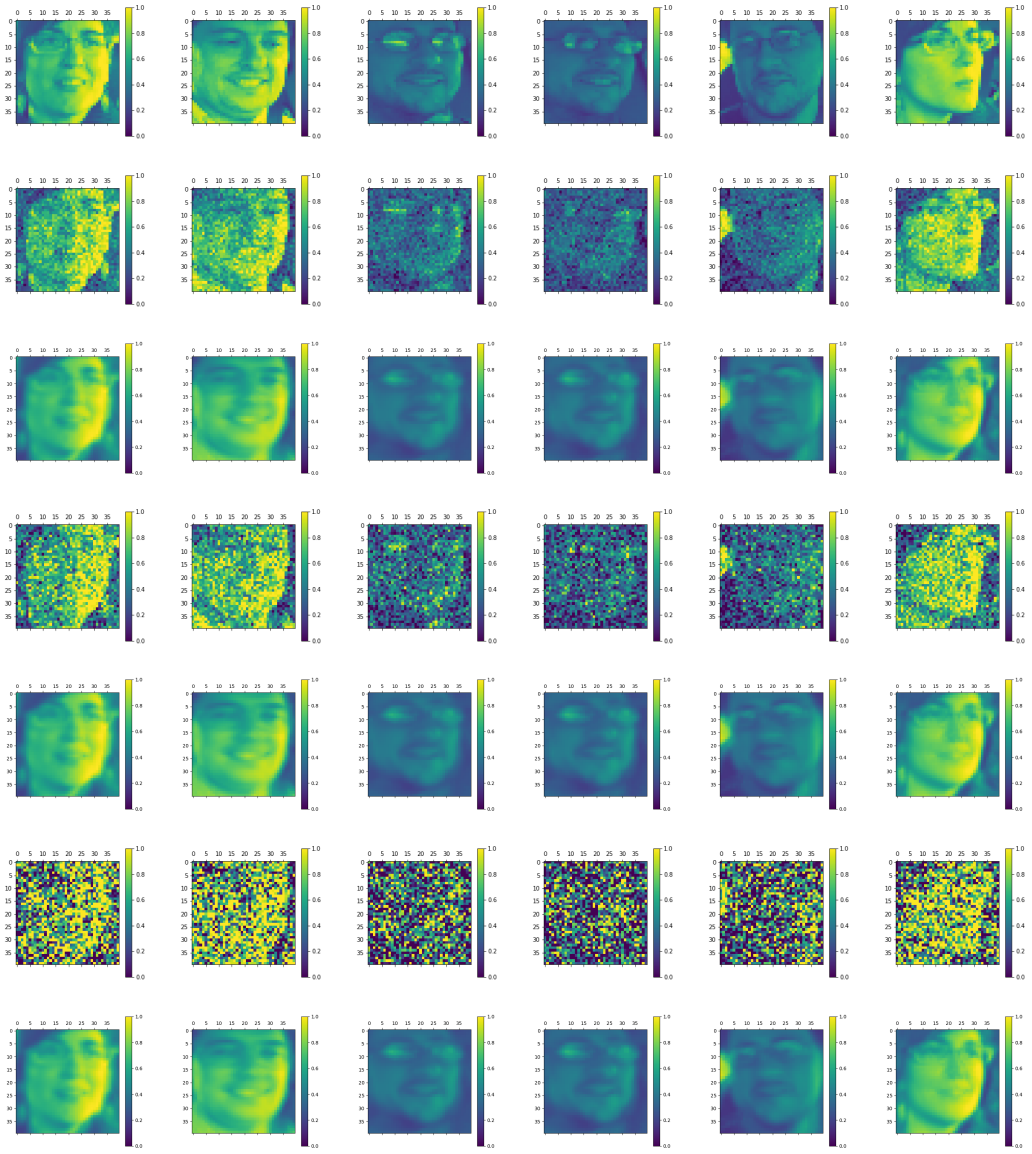

Figure 3.11: Six real images are shown in the first row. Corresponding noisy images and reconstructed images are displayed in the next two rows iteratively for three different scales $\sigma = 0.1, 0.2$ and 0.5. (Driver Face Example)

able models, the use of regularization is also justified by proving that performing VI on a sparse latent Gaussian process with this regularization is equivalent to performing VI on a related empirical Bayes model. Generally, the regularization weight λ is selected via cross validation and the weight scale depends on

data size. When cross validation is not available, we give a rule of thumb for the selection of regularization by setting $\lambda = M$ as a corresponding empirical Bayes model. We illustrate that our regularized model performs better model fitting under both ELBO_1 and ELBO_2 settings using the Anuran Calls dataset. We demonstrate the better model fitting under different latent dimension sizes Q. Moreover, we demonstrate the necessity of regularization for large datasets in noisy feature reconstruction tasks, using the Flight data. Finally, we illustrate that our regularized model also has good performance for high dimensional data such as image data. We take the Driver Face dataset for example, in which our model has better model fitting results and better reconstruction results for noisy images.

Chapter 4

Temporal Categorical Latent Gaussian Processes

This chapter proposes a novel approach to model multivariate categorical data via sparse Gaussian processes. Then we extend this approach to temporal data.

We summarize the literature of latent Gaussian process models and categorical data models in Section 4.1. It includes literature for various models as well as inference. We propose a categorical latent Gaussian process model with two regularization approaches and then we discuss its relation with priors from Beyesian perspective in Section 4.2. Moreover, we extend our model to temporal categorical latent Gaussian process model in Section 4.3 where the latent inputs are modeled by another Gaussian processes. Finally, in Section 4.4 experimental results of both synthetic data and stock index data are discussed.

4.1 Latent Gaussian Process Modeling and Categorical Data Models

In this section, we would review the literature of latent Gaussian process and categorical data models. First, we summarize the literature related to latent Gaussian process and its extensions in Section 4.1.1 and discuss categorical latent Gaussian model in Section 4.1.2. We also review the categorical latent Gaussian process model in Section 4.1.3.

4.1.1 Latent Gaussian Process Models

This section reviews the Gaussian process latent variable model (GPLVM) proposed in Lawrence (2004). It is a non-linear generalization of Probabilistic PCA. Under this framework, $\boldsymbol{Y} = (\boldsymbol{y}_1, \ldots, \boldsymbol{y}_N)^T \in \mathbb{R}^{N \times D}$ denotes N D-dimensional observations and $\boldsymbol{X} = (\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N)^T \in \mathbb{R}^{N \times Q}$ denotes N corresponding Q-dimensional latent inputs.

Then GPLVM is capable to express as D independent functions from the same Gaussian process. It shows that for each dimension d, the model is

$$\begin{aligned} \boldsymbol{Y}_{:,d} &\sim \mathcal{N}(\boldsymbol{f}_d, \sigma^2 \boldsymbol{I}) \,, \\ \boldsymbol{f}_d &= f_d(\boldsymbol{X}) \,, \\ f_d(\cdot) &\overset{i.i.d.}{\sim} GP(0, k_f(\boldsymbol{x}, \boldsymbol{x}')) \,, \end{aligned}$$

where $\boldsymbol{f}_d \in \mathbb{R}^N$ is the latent vector for the dth dimension.

Because latent variables $\boldsymbol{F} = (\boldsymbol{f}_1, \ldots, \boldsymbol{f}_D)$ are not of interest, they should be marginalized before inference. Therefore after marginalizing all latent variables,

the log marginal likelihood is given as

$$\ell = (-\frac{D}{2}) \log \det(2\pi\tilde{K}) - \frac{1}{2}\mathrm{tr}(\boldsymbol{Y}^T \tilde{K}^{-1} \boldsymbol{Y}) \tag{4.1}$$

where $\tilde{K} = K + \sigma^2 \boldsymbol{I}$ and K is the covariance matrix with respect to the latent inputs $\boldsymbol{X}$.

Taking the derivative with respect to the matrix $\tilde{K}$ on both sides of (4.1), the gradients have a closed form as

$$\frac{\partial}{\partial \tilde{K}} \ell = -\frac{D}{2} \tilde{K}^{-1} + \frac{1}{2} \tilde{K}^{-1} \boldsymbol{Y} \boldsymbol{Y}^T \tilde{K}^{-1} .$$

The definition of matrix gradient is described in Appendix A.2. Moreover, gradients of latent inputs $\boldsymbol{X}$ and gradients of hyper-parameters $\boldsymbol{\theta}$ in the covariance function are easy to derive using the chain rule.

Lawrence (2004) proposes a practical algorithm for inference in which they recursively select a subset of data as the active set using an information vector machine and then optimize the log likelihood of the active set and the log likelihood of the inactive set given the active set until all parameters converge.

In Lawrence (2004), training inference is carried out by maximizing the log marginal likelihood (4.1). In general, if we assume a prior on $\boldsymbol{X}$, $p(\boldsymbol{X}) \sim p(\boldsymbol{X}|\Phi_{\boldsymbol{X}})$, then training inference is the maximum a posteriori estimation (Ek et al., 2007).

The joint distribution of $\boldsymbol{X}$ and $\boldsymbol{Y}$ is given as

$$p(\boldsymbol{X}, \boldsymbol{Y}) = \prod_{d=1}^{D} \mathcal{N}(\boldsymbol{Y}_{:,d} | \boldsymbol{0}, \tilde{K}) p(\boldsymbol{X}|\Phi_{\boldsymbol{X}}) . \tag{4.2}$$

Latent variables $\boldsymbol{X}$ and hyper-parameters $\Phi_{\boldsymbol{X}}$ in the covariance function are

estimated by maximizing the joint distribution:

$$(\hat{\boldsymbol{X}}, \hat{\Phi}_{\boldsymbol{X}}) = \arg\max_{X, \Phi_{\boldsymbol{X}}} p(\boldsymbol{X}, \boldsymbol{Y}) .$$

On the other hand, because of $p(\boldsymbol{X}|\boldsymbol{Y}) \propto p(\boldsymbol{X}, \boldsymbol{Y})$, it is equivalent to maximizing the posterior distribution of $\boldsymbol{X}$ given observations $\boldsymbol{Y}$, suggesting:

$$(\hat{\boldsymbol{X}}, \hat{\Phi}_{\boldsymbol{X}}) = \arg\max_{\boldsymbol{X}, \Phi_{\boldsymbol{X}}} p(\boldsymbol{X}|\boldsymbol{Y}) .$$

It also implies that the training inference is equivalent to the maximum a posteriori estimation.

Different priors of latent inputs $\boldsymbol{X}$ lead to different models. Moreover, $\boldsymbol{X}$ can be modeled with unknown parameters rather than known priors. Next, we introduce some models with respect to the latent inputs $\boldsymbol{X}$.

Naive Model

The first model is the naive model. It considers a fully factorized normal prior as

$$p(\boldsymbol{X}) = \prod_{n=1}^{N} \mathcal{N}(\boldsymbol{x}_n|\boldsymbol{0}, \boldsymbol{I}_Q)$$

suggesting that each entry of latent inputs should have a identity independent standard normal distribution (Gal et al., 2015). It is widely utilized because of the computational simplicity.

Back Constraints Model

The second model is back constraints model proposed in Lawrence and Quiñonero Candela (2006). It puts constraints on latent variables $\boldsymbol{X}$ by modeling them as a smooth function of corresponding observations, meaning:

$$x_{nq} = g_q(\boldsymbol{y}_n; \boldsymbol{w}) \,,$$

where $\boldsymbol{w}$ is unknown parameters in this model.

This model utilizes a smooth mapping $g_q(\cdot)$ from $\boldsymbol{y}_n$ to $\boldsymbol{x}_n$ which guarantees the local distance preservation property. The local distance preservation means that if two latent inputs are close with each other, then their corresponding observations should be close too. It is motivated by the idea that similar observations should have similar latent inputs.

There exists numerous of mapping $g(\cdot)$. One popular mapping is based on the RBF kernel $k(\cdot, \cdot)$ and it expresses as:

$$g_q(\boldsymbol{y}_n) = \sum_{j=1}^{N} \alpha_{qj} k(\boldsymbol{y}_n, \boldsymbol{y}_j) \,,$$

where $\boldsymbol{\alpha}$ are weight parameters. The smoothness of function $g(\cdot)$ is determined by length-scale parameters in the kernel, as we discussed in (3.19). Therefore, to guarantee the sufficient smoothness of $g(\cdot)$, the length scale parameters are set as small as 10^{-6}.

On the other hand, the joint distribution (4.2) is maximized with respect to parameters $\boldsymbol{w}$ rather than latent inputs $\boldsymbol{X}$ because $\boldsymbol{X}$ is fully specified using $\boldsymbol{w}$.

Temporal Model

Temporal model is allowable to model the temporal relation across time by putting dynamical prior on latent processes. In Wang et al. (2006) and Wang et al. (2008), they utilize an auto-regressive model for the latent process, written as:

$$\boldsymbol{x}_t = f(\boldsymbol{x}_{t-1}; A) + \boldsymbol{\epsilon}\,, \tag{4.3}$$

where f is a linear combination of basis functions $f(\boldsymbol{x}; A) = \sum_i a_i \phi_i(\boldsymbol{x})$ and $\boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \sigma_a^2 \boldsymbol{I})$.

Lawrence and Moore (2007) utilize Gaussian processes to independently model each dimension of latent inputs across time. They also describe it as a special case of hierarchical GPLVM. As for each dimension q, the model for the latent process can be expressed as

$$\begin{aligned} \boldsymbol{X}_{:,q} &= g_q(\boldsymbol{t}) \\ g_q(\cdot) &\sim GP(0, k_x(t, t'))\,, \end{aligned}$$

which suggests $P(\boldsymbol{X}|\boldsymbol{t}) = \prod_{q=1}^{Q} \mathcal{N}(\boldsymbol{X}_{:,q}|\mathbf{0}, K_t)$.

4.1.2 Categorical Latent Gaussian Model

This section mainly focuses on one of the categorical data models proposed in Khan et al. (2012), which utilizes latent Gaussian model for categorical data.

As for N observations $\boldsymbol{Y} \in \mathbb{R}^{N \times D}$ (patients for example) each observation has D dimensions (different possible examinations). The dth dimension of the nth observation y_{nd} is a categorical variable with total K_d categories. For simplicity,

we assume all dimensions have the same number of levels, meaning $K_d \equiv K$ for all dimensions $d = 1, \ldots, D$.

There are different approaches to model the categorical distribution $p(\boldsymbol{Y}_{nd}|\boldsymbol{f}_{nd})$ where $\boldsymbol{f}_{nd} = (f_{nd1}, \ldots, f_{ndK})$. We will discuss those approaches later.

On the other hand, this model assumes the latent variables $\boldsymbol{f}$ are a linear function of latent inputs $\boldsymbol{X} \in \mathbb{R}^{N \times Q}$ and those inputs are modeled as N independent identically multivariate Gaussian distributions:

$$\begin{aligned} p(\boldsymbol{x}_n|\boldsymbol{\theta}) &= \mathcal{N}(\boldsymbol{x}_n|\boldsymbol{\mu}, \Sigma) \\ \boldsymbol{f}_{nd} &= W_d \boldsymbol{x}_n + \boldsymbol{w}_{0,d} \\ y_{nd} &\sim p(y_{nd}|\boldsymbol{f}_{nd}) \,. \end{aligned}$$

There are two approaches to model a categorical distribution. One is multinomial logit model and the other is multinomial probit model. Both models can be expressed in a same latent variable framework such that

$$u_{ndk} = f_{ndk} + \epsilon_{ndk} \quad \forall k = 1, \ldots, K \,,$$

where $\boldsymbol{u}$ is treated as latent variables and then the probability that the d^{th} category of the n^{th} observation belongs to category k is expressed as

$$p_{nd}(k) = p(u_{ndk} > u_{ndj}, j \neq k) \,. \tag{4.4}$$

Error ϵ is modeled by difference distribution. Different distributions refers to different models of categorical distribution. Two models are of interest in the section: multinomial logit model (MLM) and multinomial probit model (MPM).

Multinomial Logit Model

Multinomial logit model assumes that ϵ_{ndk} follow independent identically log Weibull distribution $\epsilon_{ndk} \sim EV_1(0, 1)$. After deviation, the categorical distribution is given as

$$p_{nd}(k) = \frac{e^{f_{ndk}}}{\sum_{j=1}^{K} e^{f_{ndj}}} \,. \tag{4.5}$$

This form (4.5) is also named the softmax formula in machine learning field.

Multinomial Probit Model

Multinomial probit model avoids the independent identically assumption and assumes that $\boldsymbol{\epsilon}_{nd} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{I})$. More generally it assumes $\boldsymbol{\epsilon}_{nd} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\Sigma})$. Given (4.4), the probability for n^{th} observation's d^{th} category to choose category k is written as:

$$p_{nd}(k) = \int_{R_k} p(\boldsymbol{u}_{-k}) d\boldsymbol{u}_{-k}$$

where R_k is the region where $u_k > u_j, \quad \forall k \neq j$ and $\boldsymbol{u}_{-k} = (u_1, \ldots, u_{k-1}, u_{k+1}, u_K)$.

Stick Breaking Model

Khan et al. (2012) introduce a stick breaking model as an extension of truncated stick breaking processes to generate a categorical distribution. It claims

that the distribution of the n^{th} observation and the d^{th} category is modeled as

$$\begin{aligned} p_{nd}(0) &= \sigma(f_{nd0}) \\ p_{nd}(k) &= \prod_{j \leq k-1} (1 - \sigma(f_{ndj}))\sigma(f_{ndk}), 0 < k < K \\ p_{nd}(K) &= \prod_{j=1}^{K-1} (1 - \sigma(f_{ndj})) \end{aligned}$$

where $\sigma(\cdot)$ is a logit function such that $\sigma(x) = \frac{1}{1+e^{-x}}$. Compared with the traditional stick breaking processes for Dirichlet processes $\text{DP}(\alpha, G)$, it replaces the generator variable $r_k \sim \text{Beta}(1, \alpha)$ by $\sigma(f_{ndk})$.

4.1.3 Categorical Latent Gaussian Process Model

An alternative approach to model categorical data is categorical latent Gaussian process (CLGP) model (Gal et al., 2015). It extends the linear transition between latent inputs $\boldsymbol{X}$ and latent variables $\boldsymbol{F}$ to a nonlinear transition using Gaussian processes. Moreover, M inducing inputs $\boldsymbol{Z} \in \mathbb{R}^{M \times Q}$ with corresponding variables $\boldsymbol{U} \in \mathbb{R}^{M \times D \times K}$ are introduced for sparse Gaussian processes approximation.

CLGP Model

The CLGP model is expressed as

$$\begin{aligned} x_{nq} &\sim \mathcal{N}(0, \sigma_x^2)\,, \\ \mathcal{F}_{dk}(\cdot) &\overset{iid}{\sim} \text{GP}(0, C_d(\boldsymbol{\theta}_d))\,, \\ f_{ndk} &= \mathcal{F}_{dk}(\boldsymbol{x}_n)\,, \\ y_{nd} &\sim \text{Cat}(\text{Softmax}(\boldsymbol{f}_{nd}))\,, \end{aligned} \tag{4.6}$$

where $C_d(\cdot)$ denotes a covariance function for the dth dimension of observations with hyper-parameters $\boldsymbol{\theta}_d$.

All latent inputs $\boldsymbol{X}$ are assumed to follow independent identically standard normal distribution priors across different index n and different dimension q. For each dimension d and level k, f_{ndk} is proposed to describe the relation among observations and this relation is modeled by a Gaussian process $\mathrm{GP}(C_d(\boldsymbol{\theta}_d))$ in which the corresponding hyper-parameters only depend on the category d rather than the level k for simplicity. Then observation y_{nd} is a categorical distribution modeled by a multinomial logit model. Given (4.5), the corresponding parameters come from a softmax function of $\boldsymbol{f}_{nd}$.

Under the model 4.6, there are DK independent $\boldsymbol{f}_{dk}$ and D independent Gaussian processes. The inference computation is $O(DKN^3)$ which is much expensive especially when N is large. Therefore, as for Gaussian processes $\{\mathrm{GP}(0, C_d(\boldsymbol{\theta}_d))\}_{d=1}^{D}$, Gal et al. (2015) introduces M inducing inputs $\boldsymbol{Z}$. For each dimension d and each level k, Gaussian process has its own inducing variable corresponding to $\boldsymbol{Z}$ denoted as $\boldsymbol{U}_{dk}$, suggesting that

$$u_{mdk} = \mathcal{F}_{dk}(\boldsymbol{z}_m)$$

for $d = 1, \ldots, D$ and $k = 1, \ldots, K$.

CLGP Inference

With the sparse of regression approximation (3.38), the joint marginal distribution distribution is displayed as

$$\begin{aligned} p(\boldsymbol{Y}, \boldsymbol{X}, \boldsymbol{U}) &= \int p(\boldsymbol{Y}|\boldsymbol{F}) p(\boldsymbol{F}|\boldsymbol{X}, \boldsymbol{U}) p(\boldsymbol{X}) p(\boldsymbol{U}) d\boldsymbol{F} \\ &= \prod_{n=1}^{N} \prod_{d=1}^{D} \mathrm{Cat}(y_{nd}|\mathrm{Softmax}(\boldsymbol{f}_{nd})) p(\boldsymbol{X}) p(\boldsymbol{U}) \,, \end{aligned}$$

where $\boldsymbol{f}_{dk} = K_{d;\boldsymbol{x},\boldsymbol{z}} K_{d;\boldsymbol{z},\boldsymbol{z}}^{-1} \boldsymbol{u}_{dk}$.

Then latent inputs $\boldsymbol{X}, \boldsymbol{Z}, \boldsymbol{U}$ and hyper-parameters $\boldsymbol{\theta}$ are estimated using maximum a posteriori estimation as

$$(\hat{\boldsymbol{X}}, \hat{\boldsymbol{Z}}, \hat{\boldsymbol{U}}, \hat{\boldsymbol{\theta}}) = \arg \max_{\boldsymbol{X}, \boldsymbol{Z}, \boldsymbol{U}, \boldsymbol{\theta}} p(\boldsymbol{Y}, \boldsymbol{X}, \boldsymbol{U}) \,.$$

However, the maximum a posteriori estimation only gives a point estimate rather than a posterior distribution. We introduce a variational inference of CLGP in the next section.

CLGP Variational Inference

The fully factorized variation distribution of $(\boldsymbol{X}, \boldsymbol{U}, \boldsymbol{F})$ is

$$q(\boldsymbol{X}, \boldsymbol{U}, \boldsymbol{F}) = q(\boldsymbol{X}) q(\boldsymbol{U}) q(\boldsymbol{F}) \,.$$

However, the computation is expensive when we maximize the corresponding ELBO. Therefore, another variational distribution of $(\boldsymbol{X}, \boldsymbol{U}, \boldsymbol{F})$ is proposed to generate a tractable lower bound (Damianou et al., 2016). This variational

distribution is decomposed as

$$q(\boldsymbol{X}, \boldsymbol{U}, \boldsymbol{F}) = q(\boldsymbol{X})q(\boldsymbol{U})p(\boldsymbol{F}|\boldsymbol{X}, \boldsymbol{U})\,. \tag{4.7}$$

Model inference is based on maximizing this tractable evidence lower bound (ELBO). It is given as

$$\begin{aligned}
\log p(\boldsymbol{Y}) &\geq l_{\text{elbo}} = \int q(\boldsymbol{X})q(\boldsymbol{U})p(\boldsymbol{F}|\boldsymbol{X}, \boldsymbol{U}) \log \frac{p(\boldsymbol{X})p(\boldsymbol{U})p(\boldsymbol{F}|\boldsymbol{X}, \boldsymbol{U})p(\boldsymbol{Y}|\boldsymbol{F})}{q(\boldsymbol{X})q(\boldsymbol{U})p(\boldsymbol{F}|\boldsymbol{X}, \boldsymbol{U})} d\boldsymbol{X}\boldsymbol{U}\boldsymbol{F} \\
&= l_{\text{elbo}} = \int q(\boldsymbol{X})q(\boldsymbol{U})p(\boldsymbol{F}|\boldsymbol{X}, \boldsymbol{U}) \log \frac{p(\boldsymbol{X})p(\boldsymbol{U})p(\boldsymbol{Y}|\boldsymbol{F})}{q(\boldsymbol{X})q(\boldsymbol{U})} d\boldsymbol{X}\boldsymbol{U}\boldsymbol{F} \\
&= -\text{KL}(q(\boldsymbol{X})||p(\boldsymbol{X})) - \text{KL}(q(\boldsymbol{U})||p(\boldsymbol{U})) \\
&\quad + \int q(\boldsymbol{X})q(\boldsymbol{U})p(\boldsymbol{F}|\boldsymbol{X}, \boldsymbol{U}) \log p(\boldsymbol{Y}|\boldsymbol{F}) d\boldsymbol{X}\boldsymbol{U}\boldsymbol{F}\,.
\end{aligned} \tag{4.8}$$

The variational distributions of $\boldsymbol{U}$ and $\boldsymbol{X}$ are constructed using independent normal distributions, as the same as in Titsias and Lawrence (2010).

$$\begin{aligned}
q(\boldsymbol{U}) &= \prod_{d=1}^{D} \prod_{k=1}^{K} \mathcal{N}(\boldsymbol{u}_{dk}|\boldsymbol{\mu}_{dk}, \Sigma_d)\,. \\
q(\boldsymbol{X}) &= \prod_{n=1}^{N} \prod_{q=1}^{Q} \mathcal{N}(x_{nq}|m_{nq}, s_{nq}^2)\,.
\end{aligned}$$

Since both $q(\boldsymbol{X})$ and $p(\boldsymbol{X})$ belong to the multivariate Gaussian distribution, their KL divergence has a closed-form expression as discussed in the empirical setting under Section 3.1.5, which means

$$\begin{aligned}
\text{KL}(q(\boldsymbol{X})||p(\boldsymbol{X})) &= \sum_{n=1}^{N} \sum_{q=1}^{Q} \text{KL}(q(x_{nq})||p(x_{nq})) \\
&= \sum_{n=1}^{N} \sum_{q=1}^{Q} \left(\log \frac{\sigma_x}{s_{nq}} + \frac{s_{nq}^2 + m_{nq}^2}{2\sigma_x^2} - \frac{1}{2} \right)\,.
\end{aligned}$$

The same is for the KL divergence between $q(\boldsymbol{U})$ and $p(\boldsymbol{U})$,

$$\begin{aligned}\mathrm{KL}\left(q(\boldsymbol{U})||p(\boldsymbol{U})\right) &= \sum_{d=1}^{D}\sum_{k=1}^{K}\mathrm{KL}(q(\boldsymbol{u}_{dk})||p(\boldsymbol{u}_{dk})) \\ &= \sum_{d=1}^{D}\sum_{k=1}^{K}\frac{1}{2}\left(\log\frac{|C_d(\boldsymbol{Z};\boldsymbol{\theta}_d)|}{|\Sigma_d|} - m + \mathrm{tr}(C_d(\boldsymbol{Z};\boldsymbol{\theta}_d)^{-1}\Sigma_d) + \boldsymbol{\mu}_{dk}^T C_d(\boldsymbol{Z};\boldsymbol{\theta}_d)^{-1}\boldsymbol{\mu}_{dk}\right).\end{aligned}$$

Here, $C_d(\boldsymbol{Z};\boldsymbol{\theta}_d)$ denotes the covariance matrix under the dth Gaussian process with respect to inputs $\boldsymbol{Z}$ and $C_d(\boldsymbol{Z},\boldsymbol{Z}^*;\boldsymbol{\theta}_d)$ denotes the covariance matrix under the dth Gaussian process with respect to inputs $\boldsymbol{Z}$ and $\boldsymbol{Z}^*$.

In the machine learning literature, especially in the auto-encoder literature, $\mathrm{KL}(q(\boldsymbol{X})||p(\boldsymbol{X})$ and $\mathrm{KL}(q(\boldsymbol{U})||p(\boldsymbol{U})$ are called regularization terms. They are used to minimize the distance between the variational distribution of latent variables and their prior distributions. $\int q(\boldsymbol{X})q(\boldsymbol{U})p(\boldsymbol{F}|\boldsymbol{X},\boldsymbol{U})\log p(\boldsymbol{Y}|\boldsymbol{F})d\boldsymbol{X}\boldsymbol{U}\boldsymbol{F}$ is called the reconstruction term. It is used to describe the likelihood to reconstruct the observations. Therefore, maximizing the ELBO means maximizing the reconstruction term and at the same time minimizing the distance between the variational distribution and prior distribution of latent variables.

As for the reconstruction term, directly computing the expectation is intractable. Therefore, we approximate the expectation term using Monte Carlo integration method (Gal et al., 2015). Mathematically, the integration is approximated as

$$\int q(\boldsymbol{X})q(\boldsymbol{U})p(\boldsymbol{F}|\boldsymbol{X},\boldsymbol{U})\log p(\boldsymbol{Y}|\boldsymbol{F})d\boldsymbol{X}\boldsymbol{U}\boldsymbol{F} = \frac{1}{T}\sum_{t=1}^{T}\log p(\boldsymbol{Y}|\boldsymbol{F}^{(t)}),$$

where T denotes the number of samples in Monte Carlo integration, and $\boldsymbol{F}^{(t)}$ is sampled from $p(\boldsymbol{F}|\boldsymbol{X}^{(t)},\boldsymbol{U}^{(t)})$ where both $\boldsymbol{X}^{(t)}$ and $\boldsymbol{U}^{(t)}$ are sampled from $q(\boldsymbol{X})$ and $q(\boldsymbol{U})$ respectively. Because $q(\boldsymbol{X}), q(\boldsymbol{U})$ and $p(\boldsymbol{F}|\boldsymbol{X},\boldsymbol{U})$ are all normal distributions, generating sample $\boldsymbol{F}$ is tractable.

Moreover, $p(\boldsymbol{F}|\boldsymbol{X},\boldsymbol{U}) = \prod_{d=1}^{D}\prod_{k=1}^{K} p(\boldsymbol{f}_{dk}|\boldsymbol{X},\boldsymbol{u}_{dk})$. However, it is still expensive to compute $p(\boldsymbol{f}_{dk}|\boldsymbol{X},\boldsymbol{u}_{dk})$ because it costs $O(nm^2)$. Gal et al. (2015) implicitly assumes that $\boldsymbol{f}_{dk}$ are independent conditional on inducing variables $\boldsymbol{u}_{dk}$ in training processes which is the same as the assumption in (3.47). It means

$$\begin{aligned} p(\boldsymbol{F}|\boldsymbol{X},\boldsymbol{U}) &= \prod_{d=1}^{D}\prod_{k=1}^{K}\prod_{n=1}^{N} p(f_{ndk}|\boldsymbol{x}_n,\boldsymbol{u}_{dk}) \\ &= \prod_{d=1}^{D}\prod_{k=1}^{K}\prod_{n=1}^{N} \mathcal{N}(f_{ndk}|a_{ndk},b_{ndk}^2)\,. \end{aligned} \tag{4.9}$$

where $a_{ndk} = \boldsymbol{v}_{nd}^T\Sigma_{Zd}^{-1}\boldsymbol{\mu}_{dk}$ and $b_{ndk}^2 = \sigma_n^2 - \boldsymbol{v}_{nd}^T\Sigma_{Zd}^{-1}\boldsymbol{v}_{nd}$ and $\Sigma_{Zd} = C(\boldsymbol{Z};\boldsymbol{\theta}_d), \boldsymbol{v}_{nd} = C(\boldsymbol{Z},\boldsymbol{x}_n;\boldsymbol{\theta}_d), \sigma_n^2 = C(\boldsymbol{x}_n,\boldsymbol{\theta}_d)$.

A linear transformation trick (Kingma and Welling, 2013) is introduced for sampling to improve the inference efficiency. It re-parameterizes a random variables as a function of hyper-parameters and a random variable which does not depend on hyper-parameters. Therefore, it is tractable to compute the derivative of the random variable with respect to its corresponding hyper-parameters. A general re-parameterization for the multivariate Gaussian distribution is discussed which is needed to compute the ELBO.

If random variables $\boldsymbol{x}$ follow a multivariate Gaussian distribution $\boldsymbol{x} \sim \mathcal{N}(\boldsymbol{\mu},\Sigma)$, since the covariance matrix Σ is positive definite, it can be decomposed as $\Sigma = LL^T$ where L is a lower triangular matrix. Then it can be rewritten as $\boldsymbol{x} = \boldsymbol{\mu} + \boldsymbol{L}\boldsymbol{\epsilon}$,

where $\epsilon \sim \mathcal{N}(\mathbf{0}, \boldsymbol{I})$. The corresponding derivatives are derived as

$$\begin{aligned}
\frac{\partial x_i}{\partial \mu_j} &= \begin{cases} 1 & i = j \\ 0 & i \neq j \end{cases}, \\
\frac{\partial x_i}{\partial l_{jk}} &= \begin{cases} \epsilon_k & i = j \\ 0 & i \neq j \end{cases} \qquad i \geq j\,.
\end{aligned}$$

CLGP Prediction

This section discusses CLGP prediction. In this model, hyper-parameters are $\boldsymbol{\theta}, \boldsymbol{Z}$ and variational parameters are $\boldsymbol{\mu}, \boldsymbol{\Sigma}, \boldsymbol{m}, \boldsymbol{s}$. After we get those estimates, we sequentially discuss CLGP prediction for training data, testing data and incomplete testing data.

As for training prediction, we choose the variational mean of embedding inputs $\boldsymbol{m}$ as the embedding input estimates $\hat{\boldsymbol{X}}$ and we choose the variational mean of inducing output $\boldsymbol{\mu}$ as inducing output estimates $\hat{\boldsymbol{U}}$.

Then embedding outputs $\boldsymbol{F}$ are estimated using $\hat{\boldsymbol{F}} = E(p(\boldsymbol{F}|\hat{\boldsymbol{X}}, \hat{\boldsymbol{U}}))$. Given the decomposition expression (4.9), we obtain $\hat{\boldsymbol{f}}_{ndk} = \hat{a}_{ndk}$, where $\hat{a}_{ndk} = \hat{\boldsymbol{v}}_{nd}^T \Sigma_{Zd}^{-1} \hat{\boldsymbol{u}}_{dk}$ and $\hat{\boldsymbol{v}}_{nd} = C(\boldsymbol{Z}, \hat{\boldsymbol{x}}_n; \boldsymbol{\theta}_d)$. Then the training predictive log likelihood is

$$\begin{aligned}
\ell_{training} &= \log p(\boldsymbol{Y}|\hat{\boldsymbol{F}}) \\
&= \sum_{n=1}^{N} \sum_{d=1}^{D} \log(\text{Softmax}(\hat{\boldsymbol{f}}_{nd})[y_{nd}])
\end{aligned}$$

and the training predictive log perplexity is

$$\text{p}_{training} = \frac{1}{ND} \sum_{n=1}^{N} \sum_{d=1}^{D} \log(\text{Softmax}(\hat{\boldsymbol{f}}_{nd})[y_{nd}])\,.$$

For testing prediction, assume we have $\tilde{N}$ testing observations $\tilde{\boldsymbol{Y}} = (\tilde{\boldsymbol{y}}_1, \ldots, \tilde{\boldsymbol{y}}_{\tilde{N}})$, given the inducing input $\boldsymbol{Z}$ and the inducing output estimates $\hat{\boldsymbol{U}}$, the testing embedding inputs $\tilde{\boldsymbol{X}}$ are estimated by maximizing the testing log likelihood where $\tilde{\boldsymbol{F}}$ is estimated by $E(\tilde{\boldsymbol{F}})$.

$$\begin{aligned}
\hat{\tilde{\boldsymbol{X}}} &= \arg\max_{\tilde{\boldsymbol{X}}} \ell_{testing} \\
&= \arg\max_{\tilde{\boldsymbol{X}}} \sum_{n=1}^{\tilde{N}} \sum_{d=1}^{D} \log(\text{Softmax}(\hat{\tilde{\boldsymbol{f}}}_{nd}))[\tilde{y}_{nd}] \\
&= \arg\max_{\tilde{\boldsymbol{X}}} \sum_{n=1}^{\tilde{N}} \sum_{d=1}^{D} \log(\text{Softmax}(E(\tilde{\boldsymbol{f}}_{nd})))[\tilde{y}_{nd}] \\
&= \arg\max_{\tilde{\boldsymbol{X}}} \sum_{n=1}^{\tilde{N}} \sum_{d=1}^{D} \log(\text{Softmax}(\tilde{\boldsymbol{a}}_{nd}))[\tilde{y}_{nd}],
\end{aligned}$$

where $\tilde{a}_{ndk} = \tilde{\boldsymbol{v}}_{nd}^T \Sigma_{Zd}^{-1} \boldsymbol{\mu}_{dk}$ and $\tilde{\boldsymbol{v}}_{nd} = C(\boldsymbol{Z}, \tilde{\boldsymbol{x}}_n; \boldsymbol{\theta}_d)$.

Then given the testing embedding input estimates $\hat{\tilde{\boldsymbol{X}}}$, the testing predictive log perplexity is

$$\begin{aligned}
\text{p}_{testing} &= \log p(\tilde{\boldsymbol{Y}} | \hat{\tilde{\boldsymbol{F}}}) \\
&= \frac{1}{\tilde{N}D} \sum_{n=1}^{\tilde{N}} \sum_{d=1}^{D} \log(\text{Softmax}(\hat{\tilde{\boldsymbol{f}}}_{nd})[\tilde{y}_{nd}]) \,.
\end{aligned}$$

We also discuss the situation in which testing observations are incomplete. Assume for the nth observation $\tilde{\boldsymbol{y}}_n$, only D_n observations are available. They are denoted as $\tilde{y}_{ni_{nd}}$ for $d = 1, \ldots, D_n$, where i_{nd} is a categorical index for the nth observation and the dth observable category. We also denote all categorical indexes for the n the observations as $\boldsymbol{I}_n = \{i_{nd}\}$. Under this scenario, the testing log likelihood is

$$\ell_{testing} = \sum_{n=1}^{\tilde{N}} \sum_{d \in \boldsymbol{I}_n} \log(\text{Softmax}(\hat{\tilde{\boldsymbol{f}}}_{nd})[\tilde{y}_{nd}])$$

and the testing log perplexity is

$$\mathrm{p}_{testing} = \frac{1}{\sum_{n=1}^{\tilde{N}} |\boldsymbol{I}_n|} \sum_{n=1}^{\tilde{N}} \sum_{d \in \boldsymbol{I}_n} \log(\mathrm{Softmax}(\hat{\tilde{\boldsymbol{f}}}_{nd})[\tilde{y}_{nd}]) \,.$$

We define the predictive testing log likelihood and predictive testing log perplexity as

$$\begin{aligned} \tilde{\ell}_{testing} &= \sum_{n=1}^{\tilde{N}} \sum_{d \notin \boldsymbol{I}_n} \log(\mathrm{Softmax}(\hat{\tilde{\boldsymbol{f}}}_{nd})[\tilde{y}_{nd}]) \,, \\ \tilde{\mathrm{p}}_{testing} &= \frac{1}{\sum_{n=1}^{\tilde{N}} |\boldsymbol{I}_n^c|} \sum_{n=1}^{\tilde{N}} \sum_{d \notin \boldsymbol{I}_n} \log(\mathrm{Softmax}(\hat{\tilde{\boldsymbol{f}}}_{nd})[\tilde{y}_{nd}]) \,. \end{aligned}$$

We define the testing predictive accuracy as

$$\frac{1}{\sum_{n=1}^{\tilde{N}} |\boldsymbol{I}_n^c|} \sum_{n=1}^{\tilde{N}} \sum_{d \notin \boldsymbol{I}_n} \mathbf{1}_{\arg\max(\hat{\tilde{\boldsymbol{f}}}_{nd})}(\tilde{y}_{nd})$$

where $\mathbf{1}(\cdot)$ is an indicator function.

CLGP Framework

This section discusses how to train and predict CLGP model under stochastic gradient descent (SGD) framework. The popular stochastic gradient descent methods include Adam, Adagrad and RSM methods. The framework is developed

as follows:

for $i \leftarrow 1$ **to** $N_{training}$ **do**

Update model parameters $\boldsymbol{\theta}, \boldsymbol{Z}, \boldsymbol{\mu}, \boldsymbol{\Sigma}$ given fixed embedding inputs related parameters $\boldsymbol{m}, \boldsymbol{s}$ by maximizing the ELBO;

Update embedding inputs related parameters $\boldsymbol{m}, \boldsymbol{s}$ given fixed model parameters $\boldsymbol{\theta}, \boldsymbol{Z}, \boldsymbol{\mu}, \boldsymbol{\Sigma}$ by maximizing the ELBO;

Compute the training log perplexity given $\hat{\boldsymbol{X}}, \hat{\boldsymbol{U}}, \boldsymbol{\theta}, \boldsymbol{Z}$;

end

for $j \leftarrow 1$ **to** $N_{testing}$ **do**

Update testing embedding inputs $\tilde{\boldsymbol{X}}$ given model parameters $\boldsymbol{\theta}, \boldsymbol{Z}, \boldsymbol{\mu}, \boldsymbol{\Sigma}$ by maximizing testing log likelihood;

Compute predictive testing log perplexity given $\tilde{\boldsymbol{X}}, \hat{\boldsymbol{U}}, \boldsymbol{\theta}, \boldsymbol{Z}$;

end

Algorithm 1: Training and testing framework

where $N_{training}$ denotes the number of epochs for training model and $N_{testing}$ denotes the number of epochs for training the testing embedding inputs given the trained model.

In the testing framework, because of the multimodal distribution of the testing log likelihood with respect to embedding inputs, $\tilde{\boldsymbol{X}}$ could be bad if the initial starting points are not ideal. Therefore, to get rid of the multimodal optimization issue, we propose a random-sample search approach.

In random-sample search approach, we randomly sample the initial starting points for testing embedding inputs from a standard normal distribution, which means $\tilde{X}_{nq} \sim \mathcal{N}(0, 1), \forall n, q$. Then we use the stochastic gradient decent method to estimate the optimal embedding inputs. Finally, we choose the embedding inputs which lead to the maximum predictive testing log perplexity, as the testing embedding input estimates.

4.2 Regularization on Categorical Latent Gaussian Process

As for the regularization for categorical latent Gaussian process model, two regularization approaches are proposed for inducing inputs. One is to put a penalty term for the dissimilarity between the distribution of inducing points and the distribution of embedding inputs. Another is to randomly sample M embedding inputs and treat them as inducing points. Empirical experiments show that the first approach is much better than the second approach.

4.2.1 Regularization using KL divergence

CLGP model introduces inducing inputs to simplify the computation in Gaussian process. Inducing inputs are optimized by maximizing the ELBO. However, in practice, especially when embedding inputs are unknown, inducing inputs are much closer to original point than embedding points, as can be seen in Figure 4.1. This is because $\mathrm{KL}(q(\boldsymbol{X})||p(\boldsymbol{X}))$ forces $q(\boldsymbol{X})$ closer to prior which is centered at 0 while $\boldsymbol{Z}$ cannot learn enough from maximizing ELBO. From another aspect, CLGP does not guarantee the local distribution preservation mentioned in Lawrence and Moore (2007). It is difficult for $\boldsymbol{Z}$ to allocate the important embedding locations and then the information of $\boldsymbol{Z}$ becomes useless. Therefore, after optimization, there is a significant difference between the distribution of Z and the distribution $q(\boldsymbol{X})$. It indicates that inducing inputs cannot provide enough information for the posterior of Gaussian process. In other word, it is difficult to assign suitable positions for inducing inputs in Gaussian process throughout the maximization of the ELBO.

Therefore, we introduce a regularization term in our inference to motivate

similar distributions between inducing inputs and embedding inputs. Hence it guides the assignment for inducing inputs and makes more efficient inference on embedding inputs. The regularization term is proposed as

$$\mathrm{KL}(\tilde{q}(\boldsymbol{Z})||\tilde{q}(\boldsymbol{X}),$$

where $\tilde{q}(\cdot)$ denotes variational empirical distribution of $(\cdot)$. We utilize the Gaussian distribution class for the variational distribution $\tilde{q}$.

They are naturally estimated via the sample mean and sample covariance matrix as follow:

$$\begin{aligned} \tilde{q}(\boldsymbol{Z}) &= \mathcal{N}(\hat{\boldsymbol{\mu}}_Z, \hat{\Sigma}_Z)\,, \\ \tilde{q}(\boldsymbol{X}) &= \mathcal{N}(\hat{\boldsymbol{\mu}}_X, \hat{\Sigma}_X)\,, \end{aligned}$$

where $\hat{\boldsymbol{\mu}}_Z = \bar{Z} = \frac{1}{M}\sum_{m=1}^{M} Z_m$, $\hat{\boldsymbol{\Sigma}}_Z = \frac{1}{M}\sum_{m=1}^{M}(Z_m - \bar{Z})^2$, $\hat{\boldsymbol{\mu}}_X = \bar{\boldsymbol{m}} = \frac{1}{N}\sum_{n=1}^{N}\boldsymbol{m}_{n\cdot}$ and $\hat{\Sigma}_X = \frac{1}{N}\sum_{n=1}^{N}(\boldsymbol{m}_{n\cdot} - \bar{\boldsymbol{m}})^2$.

The modified ELBO is proposed as

$$\tilde{l}_{\mathrm{elbo}} = l_{\mathrm{elbo}} - \lambda \mathrm{KL}(\tilde{q}(\boldsymbol{Z})||\tilde{q}(\boldsymbol{X})$$

where λ is a regularization weight. Usually, λ is set to be equal to the number of inducing points. Under this setting, the modified ELBO can be treated as an equivalent empirical Bayesian approach with a prior on inducing inputs. This detail is discussed in the next section.

4.2.2 Regularization using sub-sampling

Another approach to get rid of the significant different between the distribution of inducing inputs and the distribution embedding inputs is using sub-sampling approach. In detail, we randomly sample M observations and then we let $\boldsymbol{Z} = E_{q(\boldsymbol{X})}[\boldsymbol{X}_M] = \boldsymbol{m}_X$ where $\boldsymbol{X}_M$ represents embedding inputs corresponding to the M observations and $\boldsymbol{m}_X$ represents to their corresponding mean. Since inducing inputs are randomly sampled from the mean of variational distribution of embedding inputs, it will somehow shorten the distance between the distribution of inducing inputs and the distribution of embedding inputs.

4.2.3 Regularization Bayesian Theory

This section explores the underlying relation between the modified ELBO and inducing inputs' prior. Specifically, we first discuss an empirical Bayesian model with a prior for inducing points $\boldsymbol{Z}$. Then we illustrate that maximizing the modified ELBO is equivalent to maximizing a lower bound of an empirical Bayesian model.

Empirical Bayesian model with a prior of inducing inputs

Since predictive accuracy in sparse Gaussian processes strongly depends on optimal locations of inducing inputs. Reinforcing the distributions of inducing inputs and embedding inputs on a same scale is necessary. Therefore, we put an empirical prior for inducing inputs $\boldsymbol{Z}$. For the ease of computation, we assume the prior as a normal distribution and utilize the sample mean and sample covariance matrix of embedding inputs $\boldsymbol{X}$ as the estimates of the prior. Mathematically, it

suggests that

$$\begin{aligned} \boldsymbol{z}_m &\sim \mathcal{N}(\boldsymbol{z}_m|\hat{\boldsymbol{\mu}}_X, \hat{\Sigma}_X)\,, \\ q(\boldsymbol{z}_m) &\sim \mathcal{N}(\boldsymbol{z}_m|\boldsymbol{\nu}_m, \Upsilon_m)\,, \end{aligned}$$

where the empirical prior depends on the variational distribution $q(\boldsymbol{X})$. Moreover the empirical distribution is a normal approximation for the distribution of the mean of embedding inputs' variational distribution $\boldsymbol{m}$. In order to guarantee a finite variation on inducing points $\boldsymbol{Z}$, we put a constraint on $\boldsymbol{u}$ that $|\hat{\Sigma}_Z| = |\frac{\sum_{m=1}^{M}(\boldsymbol{\nu}_m - \bar{\boldsymbol{\nu}})^2}{M}| < K$.

Under this setting, the corresponding ELBO is

$$\begin{aligned} \log p(\boldsymbol{Y}) &\geq \ell_{eblo} = \int q(\boldsymbol{Z})q(\boldsymbol{X})q(\boldsymbol{U})p(\boldsymbol{F}|\boldsymbol{Z},\boldsymbol{X},\boldsymbol{U}) \log \frac{p(\boldsymbol{Z})p(\boldsymbol{X})p(\boldsymbol{U})p(\boldsymbol{Y}|\boldsymbol{F})}{q(\boldsymbol{Z})q(\boldsymbol{X})q(\boldsymbol{U})} d\boldsymbol{Z}\boldsymbol{X}\boldsymbol{U}\boldsymbol{F} \\ &= -\mathrm{KL}(q(\boldsymbol{Z})||p(\boldsymbol{Z})) - \mathrm{KL}(q(\boldsymbol{X})||p(\boldsymbol{X})) - \mathrm{KL}(q(\boldsymbol{U})||p(\boldsymbol{U})) \\ &\quad + \int q(\boldsymbol{Z})q(\boldsymbol{X})q(\boldsymbol{U})p(\boldsymbol{F}|\boldsymbol{Z},\boldsymbol{X},\boldsymbol{U}) \log p(\boldsymbol{Y}|\boldsymbol{F}) d\boldsymbol{Z}\boldsymbol{X}\boldsymbol{U}\boldsymbol{F}\,. \end{aligned}$$

Moreover, the $\mathrm{KL}(q(\boldsymbol{Z})||p(\boldsymbol{Z}))$ can be rewritten as

$$\begin{aligned} \mathrm{KL}(q(\boldsymbol{Z})||p(\boldsymbol{Z})) &= \sum_{m=1}^{M} \mathrm{KL}(q(\boldsymbol{z}_m)||p(\boldsymbol{z}_m)) \\ &= \sum_{m=1}^{M} \frac{1}{2}\left[\log \frac{|\hat{\Sigma}_X|}{|\Upsilon_m|} - Q + \mathrm{tr}(\hat{\Sigma}_X^{-1}\Upsilon_m) + (\hat{\boldsymbol{\mu}}_X - \boldsymbol{\nu}_m)^T \hat{\Sigma}_X^{-1}(\hat{\boldsymbol{\mu}}_X - \boldsymbol{\nu}_m)\right]. \end{aligned} \tag{4.10}$$

Relation between modified ELBO and empirical Bayesian model

Under the original CLGP model (4.6), for the ease of notation, we rewrite $\boldsymbol{z}_m$ by $\boldsymbol{\nu}_m$ for all $m = 1, \ldots, M$. Then, we have $\hat{\boldsymbol{\mu}}_Z = \frac{\sum_{m=1}^{M}\boldsymbol{\nu}_m}{M}$ and $\hat{\Sigma}_Z =$

$\frac{\sum_{m=1}^{M}(\boldsymbol{\nu}_m-\hat{\boldsymbol{\mu}}_Z)(\boldsymbol{u}_m-\hat{\boldsymbol{\mu}}_Z)^T}{M}$. Then to show that maximizing the modified ELBO is equivalent to maximizing the ELBO under empirical Bayesian model, we introduce two lemmas and one theorem. Lemma 1 discusses a specific family of prior of inducing inputs under fully Bayesian approach. Lemma 2 computes the regularization term. Then based on Lemma 1 and Lemma 2, we prove that maximizing the modified ELBO in original model when $\lambda = M$ is equivalent to maximizing a lower bound of the empirical Bayesian model.

Lemma 4. *Under an empirical Bayesian model with a prior of inducing points, considering a specific family of variation distributions where* $\Upsilon_m = \epsilon \boldsymbol{I}$ *for all* $m = 1, \dots M$*, log marginal likelihood satisfies the inequality:*

$$\begin{aligned}\log p(\boldsymbol{Y}) \quad \geq \quad & \ell_0 = \frac{MQ}{2}\log(\epsilon) - \frac{2}{M\mathrm{tr}(\hat{\Sigma}_X^{-1})}\epsilon - \frac{M\log K}{2} + R \\ & - \left(\frac{M}{2}\log|\hat{\Sigma}_X| - \frac{M}{2}\log|\hat{\Sigma}_Z| - \frac{MQ}{2} + \frac{1}{2}A\right) \\ & - \mathrm{KL}(q(\boldsymbol{X})||p(\boldsymbol{X})) - \mathrm{KL}(q(\boldsymbol{U})||p(\boldsymbol{U})) \end{aligned} \tag{4.11}$$

where $R = \int q(\boldsymbol{Z})q(\boldsymbol{X})q(\boldsymbol{U})p(\boldsymbol{F}|\boldsymbol{Z},\boldsymbol{X},\boldsymbol{U})\log p(\boldsymbol{Y}|\boldsymbol{F})d\boldsymbol{Z}\boldsymbol{X}\boldsymbol{U}\boldsymbol{F}$ *and* $A = \sum_{m=1}^{M}(\boldsymbol{\nu}_m - \hat{\boldsymbol{\mu}}_X)^T\hat{\Sigma}_X^{-1}(\boldsymbol{\nu}_m - \hat{\boldsymbol{\mu}}_X)$.

Proof.

$$\begin{aligned}\log p(\boldsymbol{Y}) \quad \geq \quad & \ell_{eblo} = -\mathrm{KL}(q(\boldsymbol{Z})||p(\boldsymbol{Z})) - \mathrm{KL}(q(\boldsymbol{X})||p(\boldsymbol{X})) - \mathrm{KL}(q(\boldsymbol{U})||p(\boldsymbol{U})) + R \\ = \quad & \frac{MQ}{2}\log(\epsilon) - \frac{2}{M\mathrm{tr}(\hat{\Sigma}_X^{-1})}\epsilon - \frac{M}{2}\log|\hat{\Sigma}_Z| + R \\ & - \left(\frac{M}{2}\log|\hat{\Sigma}_X| - \frac{M}{2}\log|\hat{\Sigma}_Z| - \frac{MQ}{2} + \frac{1}{2}A\right) - \mathrm{KL}(q(\boldsymbol{X})||p(\boldsymbol{X})) - \mathrm{KL}(q(\boldsymbol{U})||p(\boldsymbol{U})) \\ \geq \quad & \frac{MQ}{2}\log(\epsilon) - \frac{2}{M\mathrm{tr}(\hat{\Sigma}_X^{-1})}\epsilon - \frac{M\log K}{2} + R \\ & - \left(\frac{M}{2}\log|\hat{\Sigma}_X| - \frac{M}{2}\log|\hat{\Sigma}_Z| - \frac{MQ}{2} + \frac{1}{2}A\right) - \mathrm{KL}(q(\boldsymbol{X})||p(\boldsymbol{X})) - \mathrm{KL}(q(\boldsymbol{U})||p(\boldsymbol{U})) \end{aligned}$$

where $R = \int q(\boldsymbol{Z})q(\boldsymbol{X})q(\boldsymbol{U})p(\boldsymbol{F}|\boldsymbol{Z},\boldsymbol{X},\boldsymbol{U})\log p(\boldsymbol{Y}|\boldsymbol{F})d\boldsymbol{Z}\boldsymbol{X}\boldsymbol{U}\boldsymbol{F}$ and $A = \sum_{m=1}^{M}(\boldsymbol{\nu}_m - \hat{\boldsymbol{\mu}}_X)^T\hat{\Sigma}_X^{-1}(\boldsymbol{\nu}_m - \hat{\boldsymbol{\mu}}_X)$. □

Lemma 5. $M\mathrm{KL}(\tilde{q}(\boldsymbol{Z})||\tilde{q}(\boldsymbol{X})) = \frac{M}{2}\log|\hat{\Sigma}_X| - \frac{M}{2}\log|\hat{\Sigma}_Z| - \frac{MQ}{2} + \frac{1}{2}\sum_{m=1}^{M}(\boldsymbol{\nu}_m - \hat{\boldsymbol{\mu}}_X)^T\hat{\Sigma}_X^{-1}(\boldsymbol{\nu}_m - \hat{\boldsymbol{\mu}}_X)$.

Proof.

$$
\begin{aligned}
\mathrm{KL}(\tilde{q}(\boldsymbol{Z})||\tilde{q}(\boldsymbol{X})) &= \frac{1}{2}\left[\log\frac{|\hat{\Sigma}_X|}{|\hat{\Sigma}_Z|} - Q + \mathrm{tr}(\hat{\Sigma}_X^{-1}\hat{\Sigma}_Z) + (\hat{\boldsymbol{\mu}}_X - \hat{\boldsymbol{\mu}}_Z)^T\hat{\Sigma}_X^{-1}(\hat{\boldsymbol{\mu}}_X - \hat{\boldsymbol{\mu}}_Z)\right] \\
&= \frac{1}{2}\left[\log\frac{|\hat{\Sigma}_X|}{|\hat{\Sigma}_Z|} - Q + \mathrm{tr}\left(\hat{\Sigma}_X^{-1}((\hat{\boldsymbol{\mu}}_X - \hat{\boldsymbol{\mu}}_Z)(\hat{\boldsymbol{\mu}}_X - \hat{\boldsymbol{\mu}}_Z)^T + \hat{\Sigma}_Z)\right)\right] \\
&= \frac{1}{2}\Bigg[\log\frac{|\hat{\Sigma}_X|}{|\hat{\Sigma}_Z|} - Q + \frac{1}{M}\mathrm{tr}\Big(\hat{\Sigma}_X^{-1}(M(\hat{\boldsymbol{\mu}}_X - \hat{\boldsymbol{\mu}}_Z)(\hat{\boldsymbol{\mu}}_X - \hat{\boldsymbol{\mu}}_Z)^T \\
&\quad + \sum_{m=1}^{M}(\boldsymbol{u}_m - \hat{\boldsymbol{\mu}}_Z)(\boldsymbol{u}_m - \hat{\boldsymbol{\mu}}_Z)^T\Big)\Bigg] \\
&= \frac{1}{2}\Bigg[\log\frac{|\hat{\Sigma}_X|}{|\hat{\Sigma}_Z|} - Q + \frac{1}{M}\mathrm{tr}\Big(\hat{\Sigma}_X^{-1}(M\hat{\boldsymbol{\mu}}_X\hat{\boldsymbol{\mu}}_X^T - M\hat{\boldsymbol{\mu}}_X\hat{\boldsymbol{\mu}}_Z^T - M\hat{\boldsymbol{\mu}}_Z\hat{\boldsymbol{\mu}}_X^T + M\hat{\boldsymbol{\mu}}_Z\hat{\boldsymbol{\mu}}_Z^T \\
&\quad + \sum_{m=1}^{M}\boldsymbol{u}_m\boldsymbol{u}_m^T - (\sum_{m=1}^{M}\boldsymbol{\nu}_m)\hat{\boldsymbol{\mu}}_Z^T - \hat{\boldsymbol{\mu}}_Z(\sum_{m=1}^{M}\boldsymbol{\nu}_m)^T + M\hat{\boldsymbol{\mu}}_Z\hat{\boldsymbol{\mu}}_Z^T\Big)\Bigg] \\
&= \frac{1}{2}\left[\log\frac{|\hat{\Sigma}_X|}{|\hat{\Sigma}_Z|} - Q + \frac{1}{M}\mathrm{tr}\Big(\hat{\Sigma}_X^{-1}(\sum_{m=1}^{M}\boldsymbol{\nu}_m\boldsymbol{\nu}_m^T - M\hat{\boldsymbol{\mu}}_X\hat{\boldsymbol{\mu}}_Z^T - M\hat{\boldsymbol{\mu}}_Z\hat{\boldsymbol{\mu}}_X^T + M\hat{\boldsymbol{\mu}}_X\hat{\boldsymbol{\mu}}_X^T)\Big)\right] \\
&= \frac{1}{2}\Bigg[\log\frac{|\hat{\Sigma}_X|}{|\hat{\Sigma}_Z|} - Q + \frac{1}{M}\mathrm{tr}\Big(\hat{\Sigma}_X^{-1}(\sum_{m=1}^{M}\boldsymbol{\nu}_m\boldsymbol{\nu}_m^T - \hat{\boldsymbol{\mu}}_X(\sum_{m=1}^{M}\boldsymbol{\nu}_m)^T - (\sum_{m=1}^{M}\boldsymbol{\nu}_m)\hat{\boldsymbol{\mu}}_X^T \\
&\quad + M\hat{\boldsymbol{\mu}}_X\hat{\boldsymbol{\mu}}_X^T)\Big)\Bigg] \\
&= \frac{1}{2}\Bigg[\log\frac{|\hat{\Sigma}_X|}{|\hat{\Sigma}_Z|} - Q + \frac{1}{M}\mathrm{tr}\Big(\hat{\Sigma}_X^{-1}(\sum_{m=1}^{M}\boldsymbol{\nu}_m\boldsymbol{\nu}_m^T - \hat{\boldsymbol{\mu}}_X(\sum_{m=1}^{M}\boldsymbol{u}_m)^T - (\sum_{m=1}^{M}\boldsymbol{\nu}_m)\hat{\boldsymbol{\mu}}_X^T \\
&\quad + M\hat{\boldsymbol{\mu}}_X\hat{\boldsymbol{\mu}}_X^T)\Big)\Bigg] \\
&= \frac{1}{2}\left[\log\frac{|\hat{\Sigma}_X|}{|\hat{\Sigma}_Z|} - Q + \frac{1}{M}\mathrm{tr}\Big(\hat{\Sigma}_X^{-1}(\sum_{m=1}^{M}(\boldsymbol{\nu}_m - \hat{\boldsymbol{\mu}}_X)(\boldsymbol{\nu}_m - \hat{\boldsymbol{\mu}}_X)^T)\Big)\right] \\
&= \frac{1}{M}\sum_{m=1}^{M}\frac{1}{2}\left[\log\frac{|\hat{\Sigma}_X|}{|\hat{\Sigma}_Z|} - Q + (\boldsymbol{\nu}_m - \hat{\boldsymbol{\mu}}_X)^T\hat{\Sigma}_X^{-1}(\boldsymbol{\nu}_m - \hat{\boldsymbol{\mu}}_X)\right].
\end{aligned}
$$

Therefore, $M\mathrm{KL}(\tilde{q}(\boldsymbol{Z})||\tilde{q}(\boldsymbol{X})) = \sum_{m=1}^{M}\frac{1}{2}\left[\log\frac{|\hat{\Sigma}_X|}{|\hat{\Sigma}_Z|} - Q + (\boldsymbol{\nu}_m - \hat{\boldsymbol{\mu}}_X)^T\hat{\Sigma}_X^{-1}(\boldsymbol{\nu}_m - \hat{\boldsymbol{\mu}}_X)\right] = \frac{M}{2}\log|\hat{\Sigma}_X| - \frac{M}{2}\log|\hat{\Sigma}_Z| - \frac{MQ}{2} + \frac{1}{2}\sum_{m=1}^{M}(\boldsymbol{\nu}_m - \hat{\boldsymbol{\mu}}_X)^T\hat{\Sigma}_X^{-1}(\boldsymbol{\nu}_m - \hat{\boldsymbol{\mu}}_X)$. □

Theorem 4.2.1. Under an empirical Bayesian model with a prior of inducing points where $\Upsilon_m = \epsilon \boldsymbol{I}$ for all $m = 1, \dots M$, as $\epsilon \to 0$, maximizing its corresponding lower bound ℓ_0 is equivalent to maximizing the modified ELBO $\tilde{\ell}$ under the original CLGP model (4.6) when $\lambda = M$.

Proof. As for both bound ℓ_0 and $\tilde{\ell}_{elbo}$, parameters include hyper-parameter $\boldsymbol{\theta}, \boldsymbol{u}$ and variational parameters $\boldsymbol{\mu}, \boldsymbol{\Sigma}, \boldsymbol{m}, \boldsymbol{s}$. We denote all parameters as Θ and then given lemma 1 we have:

$$\begin{aligned}
\arg\max_{\Theta} \ell_0 &= \arg\max_{\Theta} \frac{MQ}{2}\log(\epsilon) - \frac{2}{M\mathrm{tr}(\hat{\Sigma}_X^{-1})}\epsilon - \frac{M\log K}{2} + R \\
&\quad - \Big(\frac{M}{2}\log|\hat{\Sigma}_X| - \frac{M}{2}\log|\hat{\Sigma}_Z| - \frac{MQ}{2} + \frac{1}{2}A\Big) - \mathrm{KL}(q(\boldsymbol{X})||p(\boldsymbol{X})) - \mathrm{KL}(q(\boldsymbol{U})||p(\boldsymbol{U})) \\
&= \arg\max_{\Theta} -\frac{2}{M\mathrm{tr}(\hat{\Sigma}_X^{-1})}\epsilon + R \\
&\quad - \Big(\frac{M}{2}\log|\hat{\Sigma}_X| - \frac{M}{2}\log|\hat{\Sigma}_Z| - \frac{MQ}{2} + \frac{1}{2}A\Big) - \mathrm{KL}(q(\boldsymbol{X})||p(\boldsymbol{X})) - \mathrm{KL}(q(\boldsymbol{U})||p(\boldsymbol{U}))\,.
\end{aligned}$$

As ϵ goes to 0, $\frac{2}{M\mathrm{tr}(\hat{\Sigma}_X^{-1})}\epsilon$ converges to 0 and R would degenerated to

$$\begin{aligned}
R &= \lim_{\epsilon->0} \int q(\boldsymbol{Z})q(\boldsymbol{X})q(\boldsymbol{U})p(\boldsymbol{F}|\boldsymbol{Z},\boldsymbol{X},\boldsymbol{U})\log p(\boldsymbol{Y}|\boldsymbol{F})d\boldsymbol{Z}\boldsymbol{X}\boldsymbol{U}\boldsymbol{F} \\
&= \int q(\boldsymbol{X})q(\boldsymbol{U})p(\boldsymbol{F}|\boldsymbol{Z}=\boldsymbol{u},\boldsymbol{X},\boldsymbol{U})\log p(\boldsymbol{Y}|\boldsymbol{F})d\boldsymbol{X}\boldsymbol{U}\boldsymbol{F}\,.
\end{aligned}$$

Therefore,

$$
\begin{aligned}
\arg\max_{\Theta} \lim_{\epsilon \to 0} \ell_0 &= \lim_{\epsilon \to 0} \arg\max_{\Theta} \ell_0 \\
&= \arg\max_{\Theta} \int q(\boldsymbol{X}) q(\boldsymbol{U}) p(\boldsymbol{F}|\boldsymbol{Z}=\boldsymbol{u}, \boldsymbol{X}, \boldsymbol{U}) \log p(\boldsymbol{Y}|\boldsymbol{F}) d\boldsymbol{X}\boldsymbol{U}\boldsymbol{F} \\
&\quad -\left(\frac{M}{2}\log|\hat{\Sigma}_X| - \frac{M}{2}\log|\hat{\Sigma}_Z| - \frac{MQ}{2} + \frac{1}{2}A\right) \\
&\quad -\mathrm{KL}(q(\boldsymbol{X})||p(\boldsymbol{X})) - \mathrm{KL}(q(\boldsymbol{U})||p(\boldsymbol{U})) \\
&= \arg\max_{\Theta} \int q(\boldsymbol{X}) q(\boldsymbol{U}) p(\boldsymbol{F}|\boldsymbol{Z}=\boldsymbol{u}, \boldsymbol{X}, \boldsymbol{U}) \log p(\boldsymbol{Y}|\boldsymbol{F}) d\boldsymbol{X}\boldsymbol{U}\boldsymbol{F} \\
&\quad -M\mathrm{KL}(\tilde{q}(\boldsymbol{Z})||\tilde{q}(\boldsymbol{X}) - \mathrm{KL}(q(\boldsymbol{X})||p(\boldsymbol{X})) - \mathrm{KL}(q(\boldsymbol{U})||p(\boldsymbol{U})) \\
&= \arg\max_{\Theta} \ell_{elbo} - M\mathrm{KL}(\tilde{q}(\boldsymbol{Z})||\tilde{q}(\boldsymbol{X}))\,.
\end{aligned}
$$

□

4.3 Temporal Categorical Latent Gaussian Processes Model

We propose a hierarchical model for temporal multivariate categorical processes via Gaussian processes. In Section 4.3.1, we employ latent Gaussian processes to model the nonlinear relationship between latent variables and observations and introduce inducing points to relieve the computation burden. We model multi-dimensional observations by sharing their latent variables and model the dynamics of latent process with respect to time via Gaussian processes. Model properties as well as prior selection are discussed. Section 4.3.2 proposes two variational inference approaches. It also proposes a fast approach for inference on our model with hyper-parameter priors. Moreover, a stochastic variational inference approach is proposed for large datasets. Section 4.3.3 discusses the details of model prediction.

4.3.1 Proposed Model

Suppose we have N different time series $\{\boldsymbol{y}_n\}$ with different length $T(n)$, depending on the individual index n. Observations on each time stamp are represented by a D dimensional categorical vector. y_{ndt} represents the d^{th} observation on the t^{th} time stamp of the n^{th} time series and $\tilde{t}_{nt}$ represents the t^{th} time stamp of the n^{th} time series. Each observation y_{ntd} is a categorical data with $K(d)$ classes, depending on the dimension index d. To simplify the notation, we replace $T(n)$ as T and replace $K(d)$ as K.

Our temporal categorical latent Gaussian process model includes three levels, proposed as

$$\begin{aligned} y_{ndt} &\sim \text{Cat}(\text{Softmax}(\boldsymbol{f}_{ndt}))\,, \\ f_{ndtk} &= \mathcal{F}_{dk}(\boldsymbol{x}_{nt})\,, \qquad \mathcal{F}_{dk}(\cdot) \overset{iid}{\sim} \text{GP}(0, C(\boldsymbol{\theta}_d))\,, \\ \boldsymbol{x}_{nt} &= \boldsymbol{f}_n(\tilde{t}_{nt})\,, \qquad \boldsymbol{f}_n(t) = \boldsymbol{A}\boldsymbol{\upsilon}_n(t)\,, \qquad \upsilon_{nq}(t) \overset{iid}{\sim} \text{GP}(0, C(\boldsymbol{\phi}_q))\,. \end{aligned}$$

To model categorical data y_{ndt}, we introduce the softmax function with K latent variables $\boldsymbol{f}_{ndt} = (f_{ndt1}, f_{ndt2}, \ldots, f_{ndtK})$ in the first level of model (4.12). The identifiability is achievable by fixing all $f_{ndt1} = 0$ for all $n = 1, 2, \ldots N$, $d = 1, 2, \ldots, D$ and $t = 1, 2, \ldots T$. Since modeling the correlation across the time and individuals is of interest, we introduce latent variables $\boldsymbol{X} \in \mathbb{R}^{N \times T \times Q}$. In other words, we embed f_{ndtk} into $\boldsymbol{x}_{nt}$ on a Q-dimensional latent space via a nonlinear function $\mathcal{F}_{dk}$. Within each dimension d, we give an independent identical Gaussian process prior for $\mathcal{F}_{dk}$ in the second level of model (4.12). This level is also called latent Gaussian processes because of the latent variables $\boldsymbol{X}$. Next, to model the nonlinear mappings between embedding inputs $\boldsymbol{X}$ and time $\tilde{\boldsymbol{t}}$, we utilize the linear model of coregionalization (Pelletier et al., 2004), indicating that a multivariate

function is modeled as a linear combination of independent Gaussian processes υ_{nq}. To guarantee the identification, $\boldsymbol{A}$ is specified as a lower triangular matrix with diagonal entries all positive and the variance parameter in $\boldsymbol{\phi}$ needs to be fixed as 1. If each component in $\boldsymbol{f}(t)$ is assumed to be independent and identical, then $\boldsymbol{A}$ becomes a positive diagonal matrix. This special case can be written as

$$f_{nq}(t) \overset{iid}{\sim} \mathrm{GP}(0, C(\boldsymbol{\phi}_q)), \quad \forall q = 1, \dots Q\,, \tag{4.12}$$

where $\boldsymbol{\phi}_q$ include both scale parameter and length-scale parameter. The idea of introducing additional Gaussian processes to model the latent process is firstly proposed in the hierarchical Gaussian process latent variable model (Lawrence and Moore, 2007). In the remaining of the chapter, to avoid over-parameterization, we consider the simplified modeling in (4.12).

We introduce M inducing points with inducing inputs $\boldsymbol{Z} \in \mathbb{R}^{M \times Q}$ and corresponding inducing variables $\boldsymbol{U} \in \mathbb{R}^{M \times D \times K}$ for all Gaussian processes $\mathcal{F}_{dk}(\cdot)$ to accelerate the computational speed of Gaussian processes. This approximation idea via introducing inducing points is similar to Gal et al. (2015). It implies that we have additional equations with respect to the inducing points

$$u_{mdk} = \mathcal{F}_{dk}(\boldsymbol{z}_m)\,.$$

Our model utilizes Gaussian processes for nonlinear mappings between latent variables $\boldsymbol{F}$ and embedding inputs $\boldsymbol{X}$ and nonlinear mappings between embedding inputs $\boldsymbol{X}$ and time $\tilde{\boldsymbol{t}}$. For computational convenience, we employ a squared exponential kernel for all covariance functions.

All hyper-parameters of Gaussian processes $\boldsymbol{\theta}$ and $\boldsymbol{\psi}$ are optimized via maximizing the lower bound of log likelihood.

However, modeling the hyper-parameters is expensive because there is no conjugate prior. In our case, we treat both $\boldsymbol{\theta}$ and $\boldsymbol{\psi}$ as model parameters. Since we put the GP priors on the ν_{nq} in (4.12) with 0 mean trend, our model is likely to shrink the scale of latent variables $\boldsymbol{x}$ to 0 and sequentially shrink the scale of covariance functions $C(\boldsymbol{\phi}_q)$ to 0. Also, our model may over-estimate the dependence in $C(0, \boldsymbol{\phi}_q)$. These intrinsic properties may affect the model fitting.

To address thees drawbacks of our model, there are two approaches. One approach is to give a boundary for $\boldsymbol{\psi}$ to reduce the shrinkage effects and make latent processes $v_{nq}(t)$ more predictable. The other approach is to put informative priors on hyper-parameters.

With respect to the first approach, we assume σ^2 is bounded in an interval $[0.2, 2]$ and also assume that the correlation of $v_{nq}(t)$ between any two observable consecutive time stamps is greater than 0.1 and the correlation of $v_{nq}(t)$ between any half of whole recording time is smaller than 0.1. Let $D_0 = \max(\{\tilde{t}_{ni+1} - \tilde{t}_{ni}\}_{n=1,2,\ldots N, i=1,2,\ldots,T-1})$ and $D_1 = \frac{1}{2}\min(\{\tilde{t}_{nT} - \tilde{t}_{n1}\}_{n=1,2,\ldots,N})$. Because we employ a squared exponential kernel, we have

$$\begin{aligned} \exp(-\frac{(D_0)^2}{2l^2}) > 0.1 \quad &\Rightarrow \quad l > 0.466 D_0 \,, \\ \exp(-\frac{(D_1)^2}{2l^2}) < 0.1 \quad &\Rightarrow \quad l < 0.466 D_1 \,. \end{aligned}$$

Generally, we prefer the second approach and we put suitable priors for $\boldsymbol{\phi}$. Specifically, we put priors on the latent process $\boldsymbol{\phi}_q$. Since we employ the squared exponential kernels, inverse Gamma priors are chosen on both scale parameters $\sigma^2_{\phi_q}$ and length-scale parameters $l^2_{\phi_q}$. It means that $\sigma^2_{\phi_q} \sim \text{IG}(a, b)$ and $l^2_{\phi_q} \sim \text{IG}(a, b)$, where IG refers to inverse Gamma distribution. When $a = b = 1$, $\text{IG}(a, b)$ distribution has infinite mean, this prior is considered as a weakly non-informative prior. But based on specific data, the informative priors may be preferred for a

better model regularization.

4.3.2 Variational Inference with Regularization

First of all, we derive the evidence lower bound (ELBO) of our proposed model as

$$\begin{aligned}\log p(\boldsymbol{Y}) \;\geq\; & -\mathrm{KL}(q(\boldsymbol{X})||p(\boldsymbol{X}|\tilde{\boldsymbol{t}})) - \mathrm{KL}(q(\boldsymbol{U})||p(\boldsymbol{U})) \\ & + \int q(\boldsymbol{X})q(\boldsymbol{U})p(\boldsymbol{F}|\boldsymbol{X},\boldsymbol{U}) \log p(\boldsymbol{Y}|\boldsymbol{F}) d\boldsymbol{X}\boldsymbol{U}\boldsymbol{F}\,, \end{aligned} \tag{4.13}$$

where $p(\boldsymbol{X}|\tilde{\boldsymbol{t}})$ is a product of densities of multivariate Gaussian distributions over all time series.

The variational distributions of $\boldsymbol{U}$ and $\boldsymbol{X}$ are decomposed and constructed using independent Gaussian distributions such that

$$\begin{aligned} q(\boldsymbol{U}) &= \prod_{d=1}^{D}\prod_{k=1}^{K} \mathcal{N}(\boldsymbol{u}_{dk}|\boldsymbol{\mu}_{dk}, \Sigma_d)\,, \\ q(\boldsymbol{X}) &= \prod_{n=1}^{N}\prod_{t=1}^{T}\prod_{q=1}^{Q} \mathcal{N}(x_{ntq}|m_{ntq}, s^2_{ntq})\,. \end{aligned}$$

When hyper-parameter optimization is of interest, we can introduce variational distributions $q(\boldsymbol{\phi}_q)$ and then the lower bound is redefined as

$$\begin{aligned}\log p(\boldsymbol{Y}) \;\geq\; & -\mathrm{KL}(q(\boldsymbol{\phi})||p(\boldsymbol{\phi})) + \int q(\boldsymbol{\phi})q(\boldsymbol{X}) \log\left(\frac{p(\boldsymbol{X}|\boldsymbol{\phi},\tilde{\boldsymbol{t}})}{q(\boldsymbol{X})}\right) d\boldsymbol{\phi}\boldsymbol{X} \\ & -\mathrm{KL}(q(\boldsymbol{U})||p(\boldsymbol{U})) + \int q(\boldsymbol{X})q(\boldsymbol{U})p(\boldsymbol{F}|\boldsymbol{X},\boldsymbol{U}) \log p(\boldsymbol{Y}|\boldsymbol{F}) d\boldsymbol{X}\boldsymbol{U}\boldsymbol{F}\,. \end{aligned} \tag{4.14}$$

The first integral term in (4.14) be rewritten and then be approximated using

Monte Carlo integration as

$$\begin{aligned}\int q(\boldsymbol{\phi})q(\boldsymbol{X})\log\left(\frac{p(\boldsymbol{X}|\boldsymbol{\phi},\tilde{\boldsymbol{t}})}{q(\boldsymbol{X})}\right)d\boldsymbol{\phi}\boldsymbol{X} &= E_{q(\boldsymbol{\phi})}(-\mathrm{KL}(q(\boldsymbol{X})||p(\boldsymbol{X}|\boldsymbol{\phi},\tilde{\boldsymbol{t}})) \\ &= \frac{1}{S}\sum_{s=1}^{S}(-\mathrm{KL}(q(\boldsymbol{X})||p(\boldsymbol{X}|\boldsymbol{\phi}^{(s)},\tilde{\boldsymbol{t}}))\end{aligned}$$

where S denotes the number of samples in the Monte Carlo integration and the KL divergence has a closed expression.

On the other hand, the optimal distribution of hyper-parameters $q(\boldsymbol{\phi})$ is updated by

$$q^*(\boldsymbol{\phi}) \propto p(\boldsymbol{\phi})\exp\left(E_{q(\boldsymbol{X})}\log\left(p(\boldsymbol{X}|\boldsymbol{\phi},\tilde{\boldsymbol{t}})\right)\right) .$$

Under the independence assumption across all Q latent dimensions, the update is simplified as

$$\begin{aligned}q^*(\boldsymbol{\phi}_q) &\propto p(\boldsymbol{\phi}_q)\exp\left(\sum_{n=1}^{N}E_{q(\boldsymbol{X}_{n,:,q})}\log p(\boldsymbol{X}_{n,:,q}|\boldsymbol{\phi}_q,\tilde{\boldsymbol{t}}_n)\right) \\ &\propto p(\boldsymbol{\phi}_q)\exp\left(\sum_{n=1}^{N}E_{\boldsymbol{X}_{n,:,q}}\left(-\frac{1}{2}\log(2\pi\|\Sigma_{nq}\|)-\frac{1}{2}\mathrm{tr}(\Sigma_{nq}^{-1}\boldsymbol{X}_{n,:,q}\boldsymbol{X}_{n,:,q}^T)\right)\right) \\ &\propto p(\boldsymbol{\phi}_q)\exp\left(\sum_{n=1}^{N}\log\mathcal{N}(\boldsymbol{m}_{n,:,q}|\boldsymbol{0},\Sigma_{nq})-\frac{1}{2}\mathrm{tr}\left(\Sigma_{nq}^{-1}\mathrm{diag}(\boldsymbol{s}_{n,:,q}^2)\right)\right) \quad (4.15)\end{aligned}$$

A fast approach to update $q(\boldsymbol{\phi}_q)$ is to compute the MAP of ϕ_q by maximizing

(4.15). The procedures are proposed as

for $i = 1$ **to** N_{train} **do**

Update $q(X)$, $q(U)$ and hyper-parameters $\boldsymbol{\theta}$ by maximizing (4.13) given $\boldsymbol{\phi}$;

Update $\boldsymbol{\phi}$ by maximizing (4.15) given $q(X)$, $q(U)$ and $\boldsymbol{\theta}$;

end

Algorithm 2: Fast inference approach for the model with hyper-parameter modeling.

In the lower bound (4.13), $\mathrm{KL}(q(\boldsymbol{X})||p(\boldsymbol{X}))$ and $\mathrm{KL}(q(\boldsymbol{U})||p(\boldsymbol{U}))$ are called regularization terms, minimizing the distance between the variational distributions and their prior distributions. $\int q(\boldsymbol{X})q(\boldsymbol{U})p(\boldsymbol{F}|\boldsymbol{X},\boldsymbol{U})\log p(\boldsymbol{Y}|\boldsymbol{F})d\boldsymbol{X}\boldsymbol{U}\boldsymbol{F}$ is the reconstruction term, measuring the model fitting. The standard variational inference is to maximize the ELBO, which means maximizing the reconstruction term and at the same time minimizing the distance between the variational distributions and their prior distributions for $\boldsymbol{X}$ and $\boldsymbol{U}$.

Before discussing the details of computation, we introduce a lemma as follow:

Lemma 6. *Assume the dimension size of a multivariate variable is D, and $p \sim \mathcal{N}(\boldsymbol{\mu}, \Sigma)$ and $q \sim \mathcal{N}(\tilde{\boldsymbol{\mu}}, \tilde{\Sigma})$. Then the KL divergence between p and q is*

$$\mathcal{KL}[p||q] \quad = \quad \frac{1}{2}\left(\log\frac{|\tilde{\Sigma}|}{|\Sigma|} - D + \mathrm{tr}(\tilde{\Sigma}^{-1}\Sigma) + (\tilde{\boldsymbol{\mu}} - \boldsymbol{\mu})^T\tilde{\Sigma}^{-1}(\tilde{\boldsymbol{\mu}} - \boldsymbol{\mu})\right).$$

With respect to the regularization term, because both $q(\boldsymbol{X})$ and $p(\boldsymbol{X}|\tilde{\boldsymbol{t}})$ belong to the multivariate Gaussian distribution. The KL divergence is

$$\mathrm{KL}(q(\boldsymbol{X})||p(\boldsymbol{X}|\tilde{\boldsymbol{t}})) \quad = \quad \sum_{n=1}^{N}\sum_{q=1}^{Q}\mathrm{KL}\left(q(\boldsymbol{x}_{nq})||p(\boldsymbol{x}_{nq}|\tilde{\boldsymbol{t}}_n)\right),$$

where each $\mathrm{KL}\left(q(\boldsymbol{x}_{nq})||p(\boldsymbol{x}_{nq}|\tilde{\boldsymbol{t}}_n)\right)$ has a closed-form expression using the results of KL divergence of multivariate Gaussian distributions in Lemma 6.

With Lemma 6, the KL divergence between $q(\boldsymbol{U})$ and $p(\boldsymbol{U})$ is derived as

$$\begin{aligned}\mathrm{KL}\left(q(\boldsymbol{U})||p(\boldsymbol{U})\right) &= \sum_{d=1}^{D}\sum_{k=1}^{K}\mathrm{KL}(q(\boldsymbol{u}_{dk})||p(\boldsymbol{u}_{dk})) \\ &= \sum_{d=1}^{D}\sum_{k=1}^{K}\frac{1}{2}\Bigg(\log\frac{|C(\boldsymbol{Z},\boldsymbol{Z};\boldsymbol{\theta}_d)|}{|\Sigma_d|} - m + \mathrm{tr}(C(\boldsymbol{Z},\boldsymbol{Z};\boldsymbol{\theta}_d)^{-1}\Sigma_d) + \\ &\quad \boldsymbol{\mu}_{dk}^T C(\boldsymbol{Z},\boldsymbol{Z};\boldsymbol{\theta}_d)^{-1}\boldsymbol{\mu}_{dk}\Bigg)\end{aligned}$$

where $C(\boldsymbol{Z},\boldsymbol{Z};\boldsymbol{\theta}_d)$ denotes the covariance matrix under the dth Gaussian process with respect to inputs $\boldsymbol{Z}$ and $C(\boldsymbol{Z},\boldsymbol{Z}^*;\boldsymbol{\theta}_d)$ denotes the covariance matrix under the dth Gaussian process with respect to inputs $\boldsymbol{Z}$ and $\boldsymbol{Z}^*$.

As for the reconstruction term, directly computing the expectation is intractable. Thus, the expectation term is approximated using a Monte Carlo integration method (Gal et al., 2015) or using Delta method (Wang and Blei, 2013).

Monte Carlo Method for Reconstruction Term

Mathematically, the integration is approximated by the average of samples

$$\int q(\boldsymbol{X})q(\boldsymbol{U})p(\boldsymbol{F}|\boldsymbol{X},\boldsymbol{U})\log p(\boldsymbol{Y}|\boldsymbol{F})d\boldsymbol{X}\boldsymbol{U}\boldsymbol{F} = \frac{1}{S}\sum_{s=1}^{S}\log p(\boldsymbol{Y}|\boldsymbol{F}^{(s)}) \quad (4.16)$$

where S denotes the number of samples in the Monte Carlo integration. In our case, we set $S = 5$. $\boldsymbol{F}^{(s)}$ is sampled from $p(\boldsymbol{F}|\boldsymbol{X}^{(s)},\boldsymbol{U}^{(s)})$ where both $\boldsymbol{X}^{(s)}$ and $\boldsymbol{U}^{(s)}$ are sampled from $q(\boldsymbol{X})$ and $q(\boldsymbol{U})$ respectively. Because $q(\boldsymbol{X}), q(\boldsymbol{U})$, and $p(\boldsymbol{F}|\boldsymbol{X},\boldsymbol{U})$ are all Gaussian distributions, generating the sample $\boldsymbol{F}$ is tractable.

Similar to Gal et al. (2015), we decompose the log conditional likelihood $\log p(\boldsymbol{Y}|\boldsymbol{F})$ into $\sum_{n=1}^{N}\sum_{d=1}^{D}\sum_{t=1}^{T}\log\mathrm{Softmax}(\boldsymbol{f}_{ndt})[y_{ndt}]$ and rewrite the recon-

struction term as

$$\mathrm{E}_{q(\boldsymbol{F})} \log(\boldsymbol{Y}|\boldsymbol{F}) = \sum_{n=1}^{N} \sum_{d=1}^{D} \sum_{t=1}^{T} \mathrm{E}_{q(\boldsymbol{f}_{ndt})} \log \mathrm{Softmax}(\boldsymbol{f}_{ndt})[y_{ndt}], \tag{4.17}$$

where $q(\boldsymbol{f}_{ndt}) = \prod_{k=1}^{K} q(f_{ndtk})$ according to the independent assumption in our model.

Then we derive the conditional distribution $p(f_{ndtk}|\boldsymbol{x}_{nt}, \boldsymbol{u}_{dk})$ as

$$p(f_{ndtk}|\boldsymbol{x}_{nt}, \boldsymbol{u}_{dk}) \;\; = \;\; \mathcal{N}(f_{ndtk}|a_{ndtk}, b^2_{ndtk}) \,, \tag{4.18}$$

where $a_{ntdk} = \boldsymbol{v}_{ndt}^T \Sigma_{Zd}^{-1} \boldsymbol{u}_{dk}$ and $b^2_{ntdk} = \sigma^2_{ndt} - \boldsymbol{v}_{ndt}^T \Sigma_{Zd}^{-1} \boldsymbol{v}_{ndt}$, and we specify $\Sigma_{Zd} = C(\boldsymbol{Z}, \boldsymbol{Z}; \boldsymbol{\theta}_d), \boldsymbol{v}_{ndt} = C(\boldsymbol{Z}, \boldsymbol{x}_{nt}; \boldsymbol{\theta}_d), \sigma^2_{ndt} = C(\boldsymbol{x}_{nt}, \boldsymbol{x}_{nt}; \boldsymbol{\theta}_d)$.

The linear transformation trick in (Kingma and Welling, 2013) is employed for sampling to improve the inference efficiency. It re-parameterizes a random variable as a function of two components. The first components are hyper-parameters and the second component is a random variable with no hyper-parameters. The re-parameterization makes it tractable to compute the derivative of the random variable with respect to its corresponding hyper-parameters.

The re-parameterization for multivariate Gaussion distribution is discussed in lemma 7. It is involved in the computation of the ELBO.

Lemma 7. *Suppose a random variable $\boldsymbol{x}$ follows a multivariate Gaussian distribution $\boldsymbol{x} \sim \mathcal{N}(\boldsymbol{\mu}, \Sigma)$. Since the covariance matrix Σ is positive definite, it can be decomposed as $\Sigma = LL^T$ where L is a lower triangular matrix. Then $\boldsymbol{x}$ can be re-parameterized as $\boldsymbol{x} = \boldsymbol{\mu} + \boldsymbol{L}\boldsymbol{\epsilon}$, where $\epsilon \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{I})$. The corresponding derivatives*

are derived as

$$\frac{\partial x_i}{\partial \mu_j} = \begin{cases} 1 & i = j \\ 0 & i \neq j \end{cases},$$

$$\frac{\partial x_i}{\partial l_{jk}} = \begin{cases} \epsilon_k & i = j \\ 0 & i \neq j \end{cases} \qquad i \geq j\,.$$

Delta Method for Reconstruction Term

An alternative approach to compute the reconstruction term is via Delta method. Comparing the Delta method with the Monte Carlo method, the Delta method does not require sampling and it is more robust but less accurate. To compute (4.17), it is necessary to approximate the marginal distribution $q(f_{ndtk})$ by a Gaussian distribution $\tilde{q}(f_{ndtk})$. There are two approaches.

The first approach is to approximate it by matching their mean and variance. Due to (4.18), the mean and variance of $q(f_{ndtk})$ are

$$\begin{aligned} \mathrm{E}_q(f_{ndtk}) &= \mathrm{E}_q[\mathrm{E}_p[f_{ndtk}|\boldsymbol{x}_{nt}, \boldsymbol{u}_{dk}]] \\ &= \mathrm{E}_q[\boldsymbol{v}_{ndt}^T \Sigma_{Zd}^{-1} \boldsymbol{u}_{dk}] \\ &= \langle \boldsymbol{v}_{ndt} \rangle_{q(\boldsymbol{x}_{nt})}^T \Sigma_{Zd}^{-1} \langle \boldsymbol{u}_{dk} \rangle_{q(\boldsymbol{u}_{dk})}, \end{aligned}$$

and

$$\begin{aligned} \mathrm{Var}_q(f_{ndtk}) &= \mathrm{E}_q[\mathrm{Var}_p[f_{ndtk}|\boldsymbol{x}_{nt}, \boldsymbol{u}_{dk}]] + \mathrm{Var}_q[\mathrm{E}_p[f_{ndtk}|\boldsymbol{x}_{nt}, \boldsymbol{u}_{dk}]] \\ &= \mathrm{E}_q[\sigma_{ndt}^2 - \boldsymbol{v}_{ndt}^T \Sigma_{Zd}^{-1} \boldsymbol{v}_{ndt}] + \mathrm{Var}_q[\boldsymbol{v}_{ndt}^T \Sigma_{Zd}^{-1} \boldsymbol{u}_{dk}] \\ &= \mathrm{E}_q[\sigma_{ndt}^2 - \boldsymbol{v}_{ndt}^T \Sigma_{Zd}^{-1} \boldsymbol{v}_{ndt}] + \mathrm{E}_q[\boldsymbol{v}_{ndt}^T \Sigma_{Zd}^{-1} \boldsymbol{u}_{dk} \boldsymbol{u}_{dk}^T \Sigma_{Zd}^{-1} \boldsymbol{v}_{ndt}] - \mathrm{E}_q^2[\boldsymbol{v}_{ndt}^T \Sigma_{Zd}^{-1} \boldsymbol{u}_{dk}] \\ &= \sigma_{ndt}^2 - \mathrm{tr}(\langle \boldsymbol{v}_{ndt} \boldsymbol{v}_{ntd}^T \rangle_{q(\boldsymbol{x}_{nt})} \Sigma_{Zd}^{-1}) + \mathrm{tr}(\langle \boldsymbol{v}_{ndt} \boldsymbol{v}_{ndt}^T \rangle_{q(\boldsymbol{x}_{nt})} \Sigma_{Zd}^{-1} \langle \boldsymbol{u}_{dk} \boldsymbol{u}_{dk}^T \rangle_{q(\boldsymbol{u}_{dk})} \Sigma_{Zd}^{-1}) \\ &\quad - (\langle \boldsymbol{v}_{ndt} \rangle_{q(\boldsymbol{x}_{nt})}^T \Sigma_{Zd}^{-1} \langle \boldsymbol{u}_{dk} \rangle_{\boldsymbol{q}(\boldsymbol{u}_{dk})})^2\,. \end{aligned}$$

Because of the Gaussian density $q(\boldsymbol{x}_{xt})$ and $q(\boldsymbol{u}_{dk})$, the computations of them are tractable via two summary statistics:

$$\begin{aligned}\langle \boldsymbol{v}_{ndt} \rangle_{q(\boldsymbol{x}_{nt})} &= \Psi_1^{ntd} = \int C(\boldsymbol{Z}, \boldsymbol{x}_{nt}; \boldsymbol{\theta}_d) \mathcal{N}(\boldsymbol{x}_{nt} | \boldsymbol{m}_{nt}, \mathrm{diag}(\boldsymbol{s}_{nt}^2)) d\boldsymbol{x}_{nt} \,, \\ \langle \boldsymbol{v}_{ndt} \boldsymbol{v}_{ndt}^T \rangle_{q(\boldsymbol{x}_{nt})} &= \Psi_2^{ntd} = \int C(\boldsymbol{Z}, \boldsymbol{x}_{nt}; \boldsymbol{\theta}_d) C(\boldsymbol{x}_{nt}, \boldsymbol{Z}; \boldsymbol{\theta}_d) \mathcal{N}(\boldsymbol{x}_{nt} | \boldsymbol{m}_{nt}, \mathrm{diag}(\boldsymbol{s}_{nt}^2)) d\boldsymbol{x}_{nt} \,.\end{aligned}$$

Specifically, considering squared exponential kernels with input size Q, scale parameters σ_d^2 and length-scale parameters l_d. Let $w_d = 1/l_d^2$ and the two summary statistics display as

$$\begin{aligned}(\Psi_1^{ntd})_m &= \sigma_d^2 \prod_{q=1}^{Q} \frac{\exp\left(-\frac{1}{2}\frac{w(m_{ntq} - z_{mq})^2}{w s_{ntq}^2 + 1}\right)}{(w s_{ntq}^2 + 1)^{\frac{1}{2}}} \,, \\ (\Psi_2^{ntd})_{mm'} &= \sigma_d^4 \prod_{q=1}^{Q} \frac{\exp\left(-\frac{1}{4} w (z_{mq} - z_{m'q})^2 - \frac{w(m_{ntq} - \frac{1}{2}(z_{mq} + z_{m'q}))^2}{2 w s_{ntq}^2 + 1}\right)}{(2 w s_{ntq}^2 + 1)^{\frac{1}{2}}} \,.\end{aligned}$$

The second approach is to replace both $\boldsymbol{x}_{nt}$ and $\boldsymbol{u}_{dk}$ with their mean $\boldsymbol{m}_{nt}$ and $\boldsymbol{\mu}_{dk}$. Then $\tilde{q}(f_{ndtk}) = p(f_{ndtk} | \boldsymbol{m}_{nt}, \boldsymbol{\mu}_{dk})$.

After we get the closed form for $\tilde{q}(f_{ndtk}) = \mathcal{N}(f_{ndtk} | \tilde{\mu}_{f_{ndtk}}, \tilde{\sigma}^2_{f_{ndtk}})$ parametrized by $q(\boldsymbol{X})$ and $q(\boldsymbol{U})$, we use the Delta method for inference. Due to the decomposition in (4.18) , we need to approximate $g_{ndt} = E_{\tilde{q}(\boldsymbol{f}_{ndt})} \log \mathrm{Softmax}(\boldsymbol{f}_{ndt})[y_{ndt}]$. According to (16) in Wang and Blei (2013), for any variational distribution $q(\boldsymbol{\theta}) = \mathcal{N}(\boldsymbol{\mu}, \Sigma)$ and any twice differentiable function $f(\theta)$, the objective is approximated as $E_{q(\boldsymbol{\theta})}[f(\boldsymbol{\theta})] \approx f(\boldsymbol{\mu}) + \frac{1}{2}\mathrm{tr}(\nabla^2 f(\boldsymbol{\mu})\Sigma)$. Therefore, we have

$$g_{ndt} \approx \log(S_{y_{ndt}}) + \frac{1}{2}\left(\sum_{k \neq y_{ndt}} \tilde{\sigma}^2_{f_{ndtk}} S_k S_{y_{ndt}} - \sum_{k = y_{ndt}} \tilde{\sigma}^2_{f_{ndtk}} S_k (1 - S_k)\right) = \tilde{g}_{ndt} (4.19)$$

where $S = \mathrm{Softmax}(\tilde{\boldsymbol{\mu}}_{\boldsymbol{f}_{ndt}})$. Particularly, with the second approach, $\tilde{\sigma}^2_{f_{ndtk}}$ are

independent with respect to index k. Then (4.19) is simplified as $\tilde{g}_{ndt} = \log(S_{y_{ndt}})$. Therefore, we replace all g_{ndt} with $\tilde{g}_{ndt}$ in the lower bound in (4.13).

In the case of missing data, we use Θ to describe the observable index set and denote observable $\boldsymbol{Y}$ as $\tilde{\boldsymbol{Y}} = \{Y_{ndt}\}_{(n,d,t)\in\Theta}$ with corresponding latent variables $\tilde{\boldsymbol{F}} = \{f_{ndt}\}_{(n,d,t)\in\Theta}$ and the corresponding embedding inputs as $\tilde{\boldsymbol{X}}$. The ELBO in the missing value case is expressed as

$$\begin{aligned}\log p(\tilde{\boldsymbol{Y}}) \quad \geq \quad & -\mathrm{KL}(q(\tilde{\boldsymbol{X}})||p(\tilde{\boldsymbol{X}}|\tilde{\boldsymbol{T}})) - \mathrm{KL}(q(\boldsymbol{U})||p(\boldsymbol{U})) \\ & + \int q(\tilde{\boldsymbol{X}})q(\boldsymbol{U})p(\tilde{\boldsymbol{F}}|\tilde{\boldsymbol{X}},\boldsymbol{U}) \log\left(\sum_{(n,d,t)\in\Theta} p(\boldsymbol{y}_{ndt}|\boldsymbol{f}_{ndt})\right) d\tilde{\boldsymbol{X}}\boldsymbol{U}\tilde{\boldsymbol{F}}\,.\end{aligned}$$

We propose a regularized model via maximizing the modified evidence lower bound (MELBO) which is generally studied in Chapter 3:

$$\mathrm{MELBO} = \mathrm{ELBO} - \lambda KL(\tilde{q}(\boldsymbol{Z})||\tilde{q}(\boldsymbol{X}))$$

where λ is a regularization weight.

Stochastic gradient descent (SGD) methods are employed to maximize the MELBO with respect to all parameters in our model. The details of SGD are discussed as follows. We set the number of training epochs N_{train} and evenly divide the whole dataset into N_{batch} clusters. Each cluster includes the observations $\boldsymbol{Y}_i$ and their corresponding time stamp data $\tilde{\boldsymbol{t}}_i$ and their corresponding hyper-parameters of embedding inputs, $\boldsymbol{m}_i$ for the mean and $\boldsymbol{s}_i$ for the standard deviation. In the context of the TCLGP, the model parameters include global parameters $\boldsymbol{\theta}, \boldsymbol{\phi}, \boldsymbol{Z}, \boldsymbol{\mu}, \boldsymbol{\Sigma}$ and local parameters $\boldsymbol{m}, \boldsymbol{s}$. Model inputs include both observable data and time stamps $\boldsymbol{Y}, \tilde{\boldsymbol{t}}$. Considering the robustness of our algorithm, we employ the annealing factor in Bowman et al. (2016) and propose that

$$
\begin{aligned}
\text{ELBO}(g) &= gR_0 + R_1\,, \\
R_0 &= -\text{KL}(q(\boldsymbol{X})||p(\boldsymbol{X}|\tilde{\boldsymbol{t}})) - \text{KL}(q(\boldsymbol{U})||p(\boldsymbol{U}))\,, \\
R_1 &= \int q(\boldsymbol{X})q(\boldsymbol{U})p(\boldsymbol{F}|\boldsymbol{X},\boldsymbol{U}) \log\left(\sum_{n=1}^{N}\sum_{d=1}^{D}\sum_{t=1}^{T} p(\boldsymbol{y}_{ndt}|\boldsymbol{f}_{ndt})\right) d\boldsymbol{X}\boldsymbol{U}\boldsymbol{F}\,,
\end{aligned}
$$

where R_0 and R_1 are the regularization term and the reconstruction term and anneal factor $g \in [0, 1]$).

We define $\text{MBLBO}(g) = \text{ELBO}(g) - \lambda R$. where R is the regularization term related to inducing inputs. As $g = 1$, $\text{ELBO}(g) = \text{ELBO}$ and $\text{MELBO}(g) = \text{MELBO}$. We define an annealing increase factor by $\Delta g > 0$. Specifically, $R = \text{KL}(\mathcal{N}(\hat{\boldsymbol{\mu}}_{\boldsymbol{Z}}, \hat{\Sigma}_{\boldsymbol{Z}})||\mathcal{N}(\hat{\boldsymbol{\mu}}_{\boldsymbol{m}}, \hat{\Sigma}_{\boldsymbol{m}})$, where all estimates are derived by sample mean or sample variance with respect to $\boldsymbol{Z}$ and $\boldsymbol{m}$ respectively.

As for large data, our model employs stochastic variational inference in Hoffman et al. (2013) via decomposing $\text{ELBO}(g)$ into a global term and a sum of local terms.

$$
\begin{aligned}
\text{ELBO}(g) &= -g\text{KL}(q(\boldsymbol{U})||p(\boldsymbol{U})) + \sum_{n=1}^{N}\sum_{q=1}^{Q}(-g\text{KL}(q(\boldsymbol{X}_{n,:,q})||p(\boldsymbol{X}_{n,:,q}|\tilde{\boldsymbol{t}}_n)) \\
&\quad + \sum_{n=1}^{N}\sum_{d=1}^{D}\sum_{t=1}^{T} E_{q(\boldsymbol{f}_{ndt})}p(\boldsymbol{y}_{ndt}|\boldsymbol{f}_{ndt}))\,.
\end{aligned}
$$

Considering any index $I \sim \text{Unif}(1, 2, \ldots, N)$, we define

$$
\begin{aligned}
\text{ELBO}_I(g) &= -g\text{KL}(q(\boldsymbol{U})||p(\boldsymbol{U})) + N(-g\sum_{q=1}^{Q}\text{KL}(q(\boldsymbol{X}_{I,:,q})||p(\boldsymbol{X}_{I,:,q}|\tilde{\boldsymbol{t}}_n)) \\
&\quad + \sum_{d=1}^{D}\sum_{t=1}^{T} E_{q(\boldsymbol{f}_{It})}p(\boldsymbol{y}_{Idt}|\boldsymbol{f}_{Idt}))
\end{aligned}
$$

and

$$\begin{aligned}\text{MELBO}_I(g) &= -\lambda R - g\text{KL}(q(\boldsymbol{U})||p(\boldsymbol{U})) + N(-g\sum_{q=1}^{Q}\text{KL}(q(\boldsymbol{X}_{I,:,q})||p(\boldsymbol{X}_{I,:,d}|\tilde{\boldsymbol{t}}_n)) \\ &\quad + \sum_{d=1}^{D}\sum_{t=1}^{T} E_{q(\boldsymbol{f}_{Idt})}p(\boldsymbol{y}_{Idt}|\boldsymbol{f}_{Idt}))\,. \end{aligned} \tag{4.20}$$

In the regularization term of (4.20), $\hat{\mu}_{\boldsymbol{m}}$ and $\hat{\Sigma}_{\boldsymbol{m}}$ involve local variables $\boldsymbol{m}_I$. Denote $\boldsymbol{m}_{-I}$ as all parameters of $\{\boldsymbol{m}_i\}$ except $\boldsymbol{m}_I$, and then the two terms are represented as

$$\begin{aligned}\hat{\boldsymbol{\mu}}_{\boldsymbol{m}} &= \frac{1}{NT}(\sum_{n\neq I}\sum_{t=1}^{T}\boldsymbol{m}_{nt} + \sum_{t=1}^{T}\boldsymbol{m}_{It}) \\ \hat{\Sigma}_{\boldsymbol{m}} &= \frac{1}{NT}\left(\sum_{n\neq I}\sum_{t=1}^{T}\boldsymbol{m}_{nt}\boldsymbol{m}_{nt}^T + \sum_{t=1}^{T}\boldsymbol{m}_{It}\boldsymbol{m}_{It}^T\right) - \hat{\boldsymbol{\mu}}_{\boldsymbol{m}}\hat{\boldsymbol{\mu}}_{\boldsymbol{m}}^T\,. \end{aligned} \tag{4.21}$$

It suggests that for each update of R, we only need two summary statistics $\mathcal{S}_{I1} = \sum_{n\neq I}\sum_{t=1}^{T}\boldsymbol{m}_{nt}$ and $\mathcal{S}_{I2} = \sum_{n\neq I}\sum_{t=1}^{T}\boldsymbol{m}_{nt}\boldsymbol{m}_{nt}^T$.

The algorithm is displayed as follows:

Set $g = 0$;

for $i = 1$ **to** N_{train} **do**

Sample a data index I uniformly from the data set;

Compute two summary statistics $\mathcal{S}_{I1}$ and $\mathcal{S}_{I2}$;

Given local parameters $\boldsymbol{m}_{-I}, \boldsymbol{s}_{-I}^2$, update all global parameters and local parameters $\boldsymbol{m}_I, \boldsymbol{s}_I^2$ through maximizing the MELBO(g);

$g = \min(g + \Delta g, 1)$;

end

Algorithm 3: Stochastic variational inference algorithm for large datasets.

4.3.3 Model Prediction

This section discusses the model prediction based on our variational inference. Our model has hyper-parameters $\boldsymbol{\theta}, \boldsymbol{\phi}, \boldsymbol{Z}$ and variational parameters $\boldsymbol{\mu}, \boldsymbol{\Sigma}, \boldsymbol{m}, \boldsymbol{s}$. After model training, we get their estimates denoted as $\hat{\Theta} = (\hat{\boldsymbol{\theta}}, \hat{\boldsymbol{\phi}}, \hat{\boldsymbol{Z}}, \hat{\boldsymbol{\mu}}, \hat{\boldsymbol{\Sigma}}, \hat{\boldsymbol{m}}, \hat{\boldsymbol{s}})$

We estimate $\boldsymbol{X}$ and $\boldsymbol{U}$ using their corresponding variational means $\hat{\boldsymbol{U}} = \hat{\boldsymbol{\mu}}, \hat{\boldsymbol{X}} = \hat{\boldsymbol{m}}$, Given new time stamps $\tilde{\boldsymbol{t}}^* = \{\tilde{\boldsymbol{t}}_n^*\} \in \mathbb{R}$, the corresponding embedding inputs $\hat{\boldsymbol{X}}^*$ are estimated through conditional distributions.

As for the n^{th} time series, given the estimates $\hat{\boldsymbol{x}}_n$, the posterior distribution of the latent variable $\boldsymbol{x}^*$ at time t^* is

$$\begin{aligned} p(\boldsymbol{x}^*|\hat{\boldsymbol{x}}_n) &= \prod_{q=1}^{Q} p(x_q^*|\hat{\boldsymbol{x}}_{nq}) \\ &= \prod_{q=1}^{Q} \mathcal{N}(C_0 C_1^{-1}\hat{\boldsymbol{x}}_{nq}, C(t^*, t^*; \hat{\boldsymbol{\phi}}) - C_0 C_1^{-1} C_0^T), \end{aligned} \tag{4.22}$$

where $C_0 = C(t^*, \tilde{\boldsymbol{T}}_n; \hat{\boldsymbol{\phi}}_q)$ and $C_1 = C(\tilde{\boldsymbol{T}}_n, \tilde{\boldsymbol{T}}_n; \hat{\boldsymbol{\phi}}_q)$. Then we estimate $\boldsymbol{x}^*$ using the posterior mean, $\hat{\boldsymbol{x}}^* = C_0 C_1^{-1}\hat{\boldsymbol{x}}_{nq}$.

After estimating the predictive embedding inputs $\hat{\boldsymbol{X}}^*$, their corresponding outputs are estimated by the conditional mean as $\boldsymbol{F}^*$ by $\hat{\boldsymbol{F}}^* = E(\boldsymbol{F}^*|\hat{\boldsymbol{X}}^*, \hat{\boldsymbol{U}})$. Given the similar decomposition expression of (4.18), each $\hat{f}_{ndtk}^*$ has a closed-form expression

$$\hat{f}_{ndtk}^* = \hat{a}_{ndtk}^*,$$

where $\hat{a}_{ndtk}^* = \hat{\boldsymbol{v}}_{ndt}^{*T}\hat{\Sigma}_{Zd}^{-1}\hat{\boldsymbol{\mu}}_{dk}$ and $\hat{\Sigma}_{Zd} = C(\hat{\boldsymbol{Z}}, \hat{\boldsymbol{Z}}; \hat{\boldsymbol{\theta}}_d)$, $\boldsymbol{v}_{ndt}^* = C(\hat{\boldsymbol{Z}}, \hat{\boldsymbol{x}}_{nt}^*; \hat{\boldsymbol{\theta}}_d)$.

Finally, the predictive distribution is estimated as

$$\begin{aligned} \hat{p}(\boldsymbol{Y}^*|\hat{\Theta}, \tilde{\boldsymbol{T}}^*) &= p(\boldsymbol{Y}^*|\hat{\boldsymbol{F}}^*) \\ &= \prod_{n=1}^{N}\prod_{d=1}^{D}\prod_{t=1}^{T} \text{Softmax}(\hat{\boldsymbol{f}}_{ndt}^*[y_{ndt}]), \end{aligned}$$

and the predictive perplexity is derived as

$$\begin{aligned} H &= \frac{\log p(\boldsymbol{Y}^* | \hat{\boldsymbol{F}}^*)}{|\tilde{\boldsymbol{T}}^*|} \\ &= \frac{\sum_{n=1}^{N} \sum_{d=1}^{D} \sum_{t=1}^{T^*} \log \text{Softmax}(\hat{\boldsymbol{f}}^*_{ndt}[y_{ndt}])}{NDT^*} . \end{aligned}$$

4.4 Experiments

We illustrate our model and inference on both synthetic data and real stock index data.

4.4.1 Synthetic Data for TCLGP Model

Data Description

We generate data from the TCLGP model itself. We set latent dimension size $Q = 2$, time series length $T = 20$, data dimension $D = 2$, the number of levels for the two dimensions $K = [2, 3]$ and the number of time series $N = 100$. For simplicity, we assume all time series share the same time stamps and we randomly generate 17 time stamps from a uniform distribution $\text{Unif}(0, 1)$ for testing. We choose the largest time stamp $t_{\max}$ and generate other 3 equal-space time stamps $[t_{\max} + 0.01, t_{\max} + 0.02, t_{\max} + 0.03]$ for testing. Embedding variables $\boldsymbol{X}$ on latent processes are generated from the linear coregionalization model with $\boldsymbol{A} = I$ and squared exponential kernels for $C(\boldsymbol{\phi}_q)$ where length-scale parameters are $l_1 = \exp(0), l_2 = \exp(-1)$. Then we choose squared exponential automatic relevance determination (ARD) kernels for $C(\boldsymbol{\theta}_d)$ with scale parameters $\sigma_1^2 = \sigma_2^2 = 5$ and weighted length-scale parameters $l_0 = (\exp(0), \exp(0)), l_1 = (\exp(-1), \exp(-1))$. Given embedding inputs $\boldsymbol{X}$, latent variables $\boldsymbol{F}$ and observation $\boldsymbol{Y}$ are sequentially generated.

Then under our regularization framework, we study the inference with and without priors separately.

Model without Priors

We employ $M = 20$ inducing points and set latent dimension size $Q = 2$ for inference. This section considers no prior on hyper-parameters. We implement both the Monte Carlo (MC) approach and the Delta method (DM) approach with the second approximation method for $\tilde{q}(f_{ndtk})$. We show the prediction accuracy for $n-$step forward result, $n = 1, 2, 3$ under different regularization weights $\lambda = 0, 10, 100$. The results are summarized in Table 4.1, Table 4.2 and Table 4.3 and the embedding variables X are shown in Figure 4.1.

	1-step forward	2-step forward	3-step forward
Dimension 1(MC)	0.78	0.71	0.71
Dimension 2(MC)	0.88	0.8	0.79
Predictive Perplexity(MC)	0.828	0.760	0.757
Dimension 1(DM)	0.83	0.76	0.78
Dimension 2(DM)	0.83	0.76	0.72
Predictive Perplexity(DM)	0.777	0.703	0.708

Table 4.1: Inference with $\lambda = 0$

	1-step forward	2-step forward	3-step forward
Dimension 1(MC)	0.77	0.7	0.71
Dimension 2(MC)	0.84	0.79	0.77
Predictive Perplexity(MC)	0.825	0.752	0.751
Dimension 1(DM)	0.82	0.76	0.77
Dimension 2(DM)	0.83	0.73	0.71
Predictive Perplexity(DM)	0.827	0.753	0.738

Table 4.2: Inference with $\lambda = 10$

Table 4.1, Table 4.2 and Table 4.3 demonstrate that with suitable regularization, the prediction results are improved. They also show that the Delta method

	1-step forward	2-step forward	3-step forward
Dimension 1(MC)	0.86	0.76	0.77
Dimension 2(MC)	0.87	0.77	0.78
Predictive Perplexity(MC)	0.865	0.765	0.775
Dimension 1(DM)	0.8	0.73	0.74
Dimension 2(DM)	0.84	0.76	0.71
Predictive Perplexity(DM)	0.817	0.732	0.725

Table 4.3: Inference with $\lambda = 100$

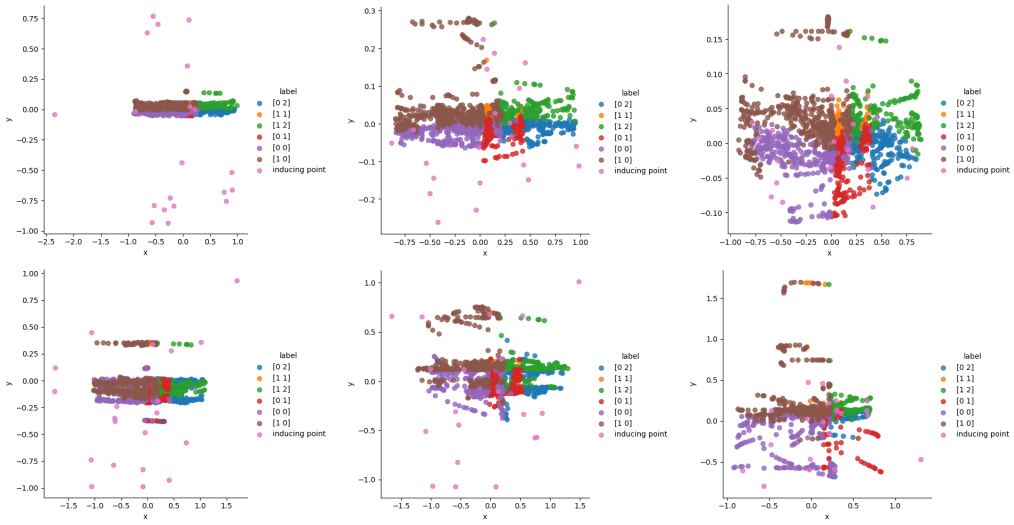

Figure 4.1: Distribution of latent variables $\boldsymbol{X}$ for different $\lambda = 0, 10, 100$.

performs similar with the Monte Carlo method. On the other hand, without priors on hyper-parameters, our inference is likely to overestimate the dependence in the latent processes.

Model with Priors

This section considers different priors on $\boldsymbol{\phi}$. As we discussed before and illustrated in Figure 4.1, our model encourages the shrinkage of the scale of latent variables. Putting priors on $\boldsymbol{\phi}$ is one way to resolve this issue. We take different priors on hyper-parameters, inference model via our proposed fast algorithm 2 and explore those model performances.

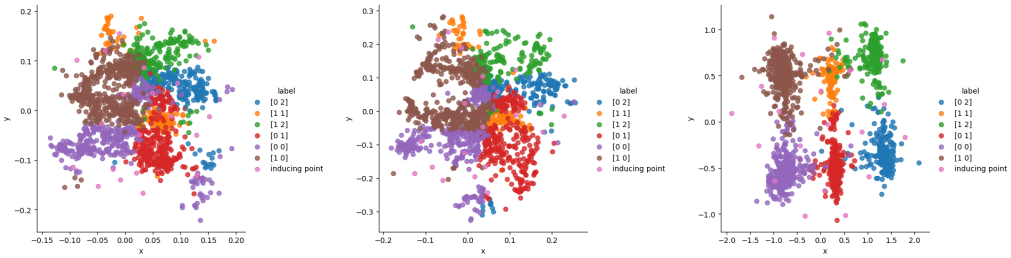

Figure 4.2: Distributions of latent variables $\boldsymbol{X}$ via standard variational inference with different priors IG$(1, 1)$ and IG$(10, 10)$ are shown on the right and middle panel. Distribution of $\boldsymbol{X}$ via stochastic variational inference with priors IG$(10, 10)$ is shown on the left panel.

In the following experiments, we consider the regularization approach with $\lambda = 100$ and take different priors on $\boldsymbol{\phi}$. Specifically, we consider priors IG$(1, 1)$ and IG$(10, 10)$. The model prediction results are displayed in Table 4.4 and the corresponding latent variables are shown in Figure 4.2. With weakly non-informative prior IG$(1, 1)$, the scale of the latent processes is likely to shrink to 0. In the contrast, with a more informative prior IG$(10, 10)$, it releases the shrinkage effects and gets better model predictions.

Moreover, we implement stochastic variantional inference in this experiments in which we consider the informative prior IG$(10, 10)$ and batch size as 10. The prediction result is shown in Table 4.4 and the embedding inputs are shown in Figure 4.2.

	1-step forward	2-step forward	3-step forward
Dimension 1(IG(1,1))	0.88	0.81	0.81
Dimension 2(IG(1,1))	0.83	0.84	0.82
Predictive Perplexity(IG(1,1))	0.841	0.793	0.790
Dimension 1(IG(10,10))	0.89	0.81	0.82
Dimension 2(IG(10,10))	0.85	0.84	0.79
Predictive Perplexity(IG(10,10))	0.838	0.802	0.793
Dimension 1(SVI, IG(10,10))	0.94	0.87	0.88
Dimension 2(SVI, IG(10,10))	0.86	0.8	0.8
Predictive Perplexity(SVI, IG(10,10))	0.889	0.820	0.805

Table 4.4: Inference on regularized models with different priors.

4.4.2 Stock Index Analysis

In the real data experiment, we select stock index data for illustration.

Data Description

We apply the TCLGP model to stock indices: SP500, Nikkei225, and DAX stock indices from 6 January 1965 to 5 December 2012 as the same as in Nicolau (2014). We utilize monthly stock indices and treat each year as an independent time series. Therefore, letting $\boldsymbol{y}_{nd}$ be monthly stock values for the nth year's and the dth stock indices, we have 48 independent time series and each time series contains three-dimensional stock indices for 12 months. Moreover, we compute the return rate for each month and evenly split all of them into 5 categories using $20\%, 40\%, 60\%, 80\%$ quantiles shown in Table 4.5. Categories 1-5 represent bear, slight bear, normal, slight bull and bull markets. For each time series, we randomly select one month in each year as testing data and treat the remaining data as training data.

Table 4.5: Percentiles of return rate for three stock indices from 1965 to 2012.

	0%	20%	40%	60%	80%	100%
SP500	-21.9	-2.5	-0.1	1.8	3.7	16.5
Nikkei225	-19.8	-3.9	-0.5	2.2	4.8	20.1
DAX	-23.4	-3.5	-0.4	2.1	5	21.9

Hyper-parameters Analysis

We carry out a bunch of experiments with $\lambda = 100$ using different settings of hyper-parameters and then show their MELBO, training and testing predictive accuracy and corresponding mean absolute difference.

Predictive accuracy and mean absolute difference are defined for both training

data and testing data as

$$\begin{aligned}
\text{TRPA} &= \frac{\sum_{n=1}^{N}\sum_{t=1}^{T-1}\mathbf{1}(\boldsymbol{y}_{n\cdot t}^{train}=\hat{\boldsymbol{y}}_{n\cdot t}^{train})}{N(T-1)}, \\
\text{TRMDA} &= \frac{\sum_{n=1}^{N}\sum_{d=1}^{D}\sum_{t=1}^{T-1}|y_{ndt}^{train}-\hat{y}_{ndt}^{train}|}{NDT}, \\
\text{TPA} &= \frac{\sum_{n=1}^{N}\mathbf{1}(\boldsymbol{y}_{n\cdot}^{test}=\hat{\boldsymbol{y}}_{n\cdot}^{test})}{N}, \\
\text{TMDA} &= \frac{\sum_{n=1}^{N}\sum_{d=1}^{D}|y_{nd}^{test}-\hat{y}_{nd}^{test}|}{ND}.
\end{aligned}$$

On the other hand, we set the same scale parameter ϕ_{σ^2} and the same scale length parameter ϕ_l for all hyper-parameters $\{\boldsymbol{\phi}_q\}$. Considering $\phi_{\sigma^2} = 0.5, 2$ and $\phi_l = 0.01, 0.05, 0.1$, experiment results are shown in Table 4.6.

Table 4.6 illustrates that choosing appropriate hyper-parameters ϕ_σ^2, ϕ_l is important. When we fix ϕ_σ^2, a smaller ϕ_l would cause an over-fitting issue on the latent processes while a bigger ϕ_l would weaken the clusters on the latent space and make categorical mapping difficult.

Table 4.6: MELBO, predictive accuracy and mean absolute difference for training data and testing data under different settings of hyper-parameters on latent Gaussian processes.

ϕ_{σ^2}	0.5			2		
ϕ_l	0.01	0.05	0.1	0.01	0.05	0.1
MELBO	-2698.79	-2671.42	-2947.10	-2691.02	-2710.76	-2920.42
TRPA	0.69	0.70	0.71	0.72	0.71	0.71
TRMAD	0.45	0.42	0.49	0.46	0.48	0.50
TPA	0.21	0.24	0.18	0.21	0.21	0.18
TMAD	1.53	1.45	1.65	1.53	1.53	1.65

Latent Processes Visualization

We choose the hyper-parameters $\phi_{\sigma^2} = 0.5$ and $\phi_l = 0.05$. Denote predictive categories given a certain latent space $\boldsymbol{x}_{pred}$ as $\boldsymbol{y}_{pred} \in [0, \ldots, 4]^3$. Next we sum

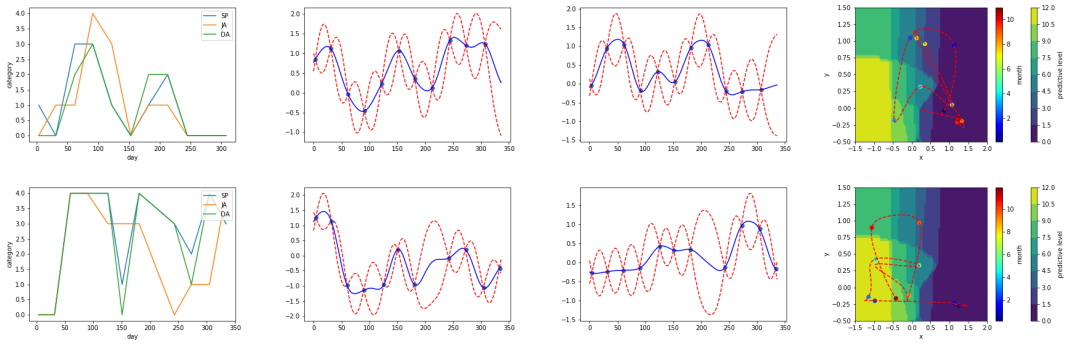

Figure 4.3: Categorical plot (left) and predictive posterior latent process (middle) and estimated latent tracing (right) in 2008 and 2009.

up $\boldsymbol{y}_{pred}$ denoted as $\tilde{y}_{pred} = \sum \boldsymbol{y}_{pred} \in [0, \ldots, 12]$. Then we plot a contour using $(\boldsymbol{x}_{pred}, \tilde{y}_{pred})$ on the latent space shown in Figure 4.3. On the surface, a high value indicates a bull market while a low value indicates a bear market. Then we show predictive posterior latent process as well as estimated latent traces in 2008 and 2009 in Figure 4.3. The estimated latent traces show that the estimated latent trace for 2008 goes through almost all dark colored areas while that for 2009 goes through almost only light colored areas. The contrast exactly matches the fact that a financial crisis happened in 2008 leading the US stock market to a bear market while the US economy returned to normal in 2009.

4.5 Conclusion

This chapter proposes a hierarchical model for multivariate categorical processes. We employ latent Gaussian process to model the nonlinear relation between latent varibles and observations and introduce inducing point to relieve the computation burden. We model the multi-dimensional data by sharing their latent variables and model the latent processes via other Gaussian processes. For inference, we propose two variantional inference approaches based on Monte Carlo

method and Delta method separately. We also provide a fast algorithm for model inference with hyper-parameter priors and a stochastic vcariational inference for large datasets. Our model and inferences are illustrated on both synthetic data and real stock index data.

Chapter 5

Nonstationary Multivariate Gaussian Processes

In this chapter, we propose a class of nonstationary multivariate Gaussian process models for electronic health records (EHR). It is able to jointly model time varying clinical variables, where the key parameters including length-scales, standard deviations and the correlations between the observed output, are all time dependent.

Since we already discussed the existing literature of EHR and relevant models in Section 1.1.2, we directly discuss our model. Our proposed nonstationary multivariate Gaussian processes (NMGP) are explored in Section 5.1. In particular, we propose one computationally efficient model, which is a special case of our NMGP. The relation between our both models and existing models are explored. In Section 5.2, we propose inference and prediction approaches for both proposed models with fully Bayesian approach via Hamiltonian Monte Carlo (HMC) and an approximate Bayesian approach via maximum a posteriori (MAP). In Section 5.3, we validate our model on synthetic data as well as on electronic health records (EHR) data from Kaiser Permanente (KP). We show that the proposed

model provides better predictive performance over a stationary model as well as uncovers interesting latent correlation processes across vitals which potentially contributes to early clinical detection.

5.1 Nonstationary Multivariate Gaussian Processes

Inpatient clinical time series data are composed of measurements of multiple correlated patient clinical variables or vital signs. Furthermore, the statistical properties of the data may not be constant across time due to physiological changes from acute disease onset. For these reasons, we propose a nonstationary multivariate Gaussian process (NMGP) to model EHR data. Importantly, it is the first such multivariate Gaussian process to model both nonstationarity in the length-scale parameter, signal variances and the correlation matrix across observed clinical variables.

We briefly review some basic properties of both multivariate Gaussian processes in the following sections. We first introduce the background of multivariate Gaussian process in Section 5.1.1, In Section 5.1.2 we present our novel generalized nonstationary multivariate Gaussian process model in details.

5.1.1 Background

A Multivariate Gaussian process (MGP) defines a distribution over multivariate functions $\boldsymbol{f}(t) = (f_1(t), \dots f_M(t))^T$. For any collection of time $\boldsymbol{t} = (t_1, \dots, t_N)^T$, the function values $\boldsymbol{f}_n = \boldsymbol{f}(t_n)$ follow a multivariate normal distribution

$$\vec{\boldsymbol{f}} \equiv \text{vec}([\boldsymbol{f}_1, \dots, \boldsymbol{f}_N]^T) \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{K}^f),$$

where vec is the vectorization operator and the covariance matrix K^f is nonnegative definite, generated from a covariance function K^f such that for any two time inputs $t, t' \in \{t_1, ..., t_N\}$ and any two dimensions $m, m' \in \{1, ..., M\}$, the covariance between the values $\boldsymbol{f}(t)[m]$ and $\boldsymbol{f}(t')[m']$ is given by $K^f(t, t', m, m')$ and covariance between $\boldsymbol{f}(t)$ and $\boldsymbol{f}(t')$ is given by $K^f(t, t')$. A MGP is said to be separable when a decomposition exists such that

$$K^f(t, t', m, m') = B_{m,m'} K(t, t'), \tag{5.1}$$

for a covariance matrix $\boldsymbol{B}$ of the dimensions only and a correlation function K of the time only. In matrix notation this is equivalent to $\boldsymbol{K}^f = \boldsymbol{B} \otimes \boldsymbol{K}$, where the covariance matrix $\boldsymbol{B} \in \mathbb{R}^{M \times M}$ summarizes the relations across output dimensions, the correlation matrix $\boldsymbol{K} = K(\boldsymbol{t}, \boldsymbol{t})$ summarizes the relations across time inputs, and $\otimes$ denotes the Kronecker product. In the past, separable cross-covariance structures were called intrinsic coregionalization (Helterbrand and Cressie, 1994). Mardia and Goodall (1993) employ the separability to model spatio-temporal data and Bhat et al. (2010) utilize it for computer model calibration.

The most common approach to building cross-covariance functions is by combining univariate covariance functions. Three primary options are the linear model of coregionalization (Bourgault and Marcotte, 1991; Goulard and Voltz, 1992), various convolution techniques (Ver Hoef and Barry, 1998; Ver Hoef et al., 2004; Gneiting et al., 2010) and use of latent dimensions (Apanasovich and Genton, 2010).

The most popular construction approach is the linear model of coregionalization (LMC) (Bourgault and Marcotte, 1991; Goulard and Voltz, 1992), because it is easy to interpret. The cross covariance function is a linear combination of R

independent univariate correlation functions, taking the form

$$K^f(t,t') = \sum_{r=1}^{R} K_r(t,t') L_r L_r^T, \tag{5.2}$$

for an integer $1 \leq R \leq M$, where $K_r(\cdot)$ are valid correlation functions and L_r is the r^{th} column of $\boldsymbol{L} \in \mathbb{R}^{M \times R}$. When $R = 1$ or when employing the same corelation function $K_r(\cdot,\cdot) \equiv K(\cdot,\cdot)$, the cross covariance function is separable as in (5.1). The LMC has been extended to the nonstationary setup by considering nonstationary univariate correlation $K_r(\cdot,\cdot)$ or allowing the coefficients to be spatially varying (Gelfand et al., 2004) such that

$$K^f(t,t') = \sum_{r=1}^{R} K_r(t,t') L_r(t) L_r^T(t') \,.$$

Gelfand et al. (2004) indirectly model $B(t) = L(t)L(t)^T$ via a matrix-variate inverse Wishart spatial process. However, this approach is prohibitive for the size of data we encountered. A simpler option is to model $L(t)$ directly (Guhaniyogi et al., 2013) using independent Gaussian processes.

5.1.2 Generalized Nonstationary Multivariate Gaussian Processes

We now present our Generalized Nonstationary Multivariate Gaussian Processes (GNMGP) in detail. The hierarchical representation of the model is as follows:

$$
\begin{aligned}
\boldsymbol{y}(t) &= L(t)\boldsymbol{g}(t) + \boldsymbol{\epsilon}(t), \\
\boldsymbol{\epsilon}(t) &\sim \mathcal{N}(0, \sigma^2_{err} I), \\
g_d(t) &\overset{iid}{\sim} \mathrm{GP}(0, K) \qquad d = 1, 2, ..., M, \\
L_{ij}(t) &\sim \mathrm{GP}(0, K_L) \qquad i > j\,, \\
\log(L_{ij})(t) &\sim \mathrm{GP}(0, K_L) \qquad i = j\,, \\
\sigma^2_{err} &\sim \mathrm{IG}(a, b)\,,
\end{aligned} \tag{5.3}
$$

where IG denotes the inverse Gamma distribution and $\boldsymbol{\epsilon}(t) = (\epsilon_1(t), \ldots, \epsilon_M(t))^T$ are observation noise. Heteroscedasticity of each dimension is modeled by putting a GP prior on $\epsilon_m(t)$ in Heinonen et al. (2016), while we model $\epsilon_m(t)$ via identical independent Wiener processes with the same variance σ^2_{err} for parsimony and we focus on the flexible modeling of the underlying multivariate process $\boldsymbol{f}(t) = L(t)\boldsymbol{g}(t)$. In order to guarantee model identifiability, at any time t, $L(t)$ is a lower triangular matrix with strictly positive diagonal entries, which models the varying variance and correlation across the dimensions and K is a correlation function modeling the smoothness across time inputs. Therefore, we employ independent Gaussian processes to model off-diagonal entries $L_{ij}(t), i > j$ and other independent Gaussian processes to model the on-diagonal entries $L_{ij}(t), i = j$ as in Guhaniyogi et al. (2013). To model the varying changing rate, we utilize a Gibbs kernel for the independent GPs $g_d(t)$, where nonstationarity is achieved by

placing a GP prior on the log length-scale process,

$$K(t,t') = \sqrt{\frac{2\ell(t)\ell'(t)}{\ell(t)^2+\ell(t')^2}}\exp\left(-\frac{(t-t')^2}{\ell(t)^2+\ell(t')^2}\right)$$
$$\ell(t) \sim \text{logGP}(0, K_{\tilde{\ell}})$$

where logGP refers to log Gaussian process. Finally, hyperparameters of the covariance function $K_{\tilde{\ell}}$ can be chosen appropriately for the application.

Because we consider the same correlation function $K(t,t')$ for all dimension d, the cross covariance function of the resulting underlying multivariate process $\boldsymbol{f}(t) = L(t)\boldsymbol{g}(t)$ is given by

$$\begin{aligned} K^f(t,t') &= L(t)\text{cov}(\boldsymbol{g}(t),\boldsymbol{g}(t'))L^T(t') \\ &= K(t,t')L(t)L^T(t'). \end{aligned} \tag{5.4}$$

The proposed GNMGP model is understood as generalizations of existing GP models. Comparing with the spatial varying LMC in Gelfand et al. (2004), our model emphasizes the varying change rate via time-dependent length-scale. Comparing with the nonstationary model in Heinonen et al. (2016), our model extends the univariate kernel to multivariate via the spatial varying linear model of coregionalization.

This general model is nonseparable, meaning the covariance function cannot be decomposed into components that are functions of either time or dimension, but not both. We also propose a special case of the above model, which is separable and available to relieve the computational burden via kronecker algebra. We name it as separable nonstationary multivariate Gaussian processes (SNMGP) model. The SNMGP is modeled by decomposing $L(t)$ by $L(t) = \sigma(t)L$, for some

positive function $\sigma(t)$ and constant lower-triangular matrix L with strict positive number on diagonal. This case is called the separable case, because the cross-covariance function is separable. In detail, letting $B = LL^t$ we see from (5.4), that the GNMGP kernel function reduces to a separable cross-covariance function, $K^f(t,t') = \sigma(t)\sigma(t')K(t,t')B$. To finish the specification of this model we assume $\log(\sigma(t)) \sim \mathrm{GP}(0, K_{\tilde{\sigma}})$, $L_{ij} \sim \mathcal{N}(0,c)$ for $i > j$ and $\log(L_{ij}) \sim \mathcal{N}(0,c)$ for $i = j$. As before, all hyperparameters of the model need to be chosen accordingly to the specific application. We also note that when we treat $L(t) \equiv \boldsymbol{L}$ and consider a stationary kernel $K(t,t')$, this special case is a vanilla linear model of coregionalization (LMC) (Bourgault and Marcotte, 1991; Goulard and Voltz, 1992), whose cross-covariance function is stationary.

Finally, we note that different correlation functions are available for $K(t,t')$. For example, Paciorek and Schervish (2004) proposed a class of nonstationary kernels for univariate output with a Matérn kernel. In this paper, we employ a Gibbs kernel for varying smoothness.

5.2 Inference

We propose two inference approaches, maximum a posteriori (MAP) and fully Bayesian inference. This section discusses the case of complete data, which means at each location or time stamp t, all observations $\boldsymbol{y}_t$ are available. Inference for incomplete data, where not all $\boldsymbol{y}_t$ are available at each t, follows from standard Gaussian process methods for marginalizing over missing data (Rasmussen and Williams, 2005). Note that for ease of exposition we introduce the following notation: $\tilde{\ell}(t) \equiv \log(\ell(t))$ and $\tilde{\sigma}(t) \equiv \log(\sigma(t))$.

5.2.1 Maximum A Posteriori (MAP)

This section considers maximum a posteriori inference for both models.

In the separable case, model parameters $\sigma^2_{err}, \boldsymbol{L}, \tilde{\boldsymbol{\ell}}, \tilde{\boldsymbol{\sigma}}$ are of interest. The marginal posterior of these parameters are

$$\begin{aligned} p(\sigma^2_{err}, \boldsymbol{L}, \tilde{\boldsymbol{\ell}}, \boldsymbol{\sigma}|\boldsymbol{y}, \boldsymbol{t}) &= \int p(\boldsymbol{f}, \sigma^2_{err}, \boldsymbol{L}, \boldsymbol{g}, \tilde{\boldsymbol{\ell}}, \boldsymbol{\sigma}|\boldsymbol{y}, \boldsymbol{t}) d\boldsymbol{g} d\boldsymbol{f} \\ &\propto \mathcal{N}(\vec{\boldsymbol{y}}|\boldsymbol{0}, \boldsymbol{K}^f + \sigma^2_{err}\boldsymbol{I}) \mathcal{N}(\tilde{\boldsymbol{\sigma}}|\boldsymbol{0}, \boldsymbol{K}_{\tilde{\boldsymbol{\sigma}}}) \mathcal{N}(\tilde{\boldsymbol{\ell}}|\boldsymbol{\mu}_{\tilde{\boldsymbol{\ell}}}, \boldsymbol{K}_{\tilde{\boldsymbol{\ell}}}) \mathrm{IG}(\sigma^2_{err}|a, b) \\ &\prod_{i>j} \mathcal{N}(L_{ij}|0, c) \prod_i \mathcal{N}(\log L_{ii}|0, c) \left| \frac{d \log L_{ii}}{d L_{ii}} \right|. \end{aligned} \tag{5.5}$$

The most expensive computation comes from $\mathcal{N}(\boldsymbol{y}|\boldsymbol{0}, \boldsymbol{K}^f + \sigma^2_{err}\boldsymbol{I})$. Since this setting is separable $\boldsymbol{K}^f = \boldsymbol{B} \otimes \boldsymbol{K}$, methods exploiting Kronecker structure Saatçi (2012); Wilson (2014) are discussed. Directly computing the likelihood costs $O(M^3N^3)$, due to the computation of $(\boldsymbol{B} \otimes \boldsymbol{K} + \sigma^2_{err}\boldsymbol{I})^{-1}$ and $\log\det(\boldsymbol{B} \otimes \boldsymbol{K} + \sigma^2_{err}\boldsymbol{I})$.

Efficient computation approaches for the two terms are proposed through eigen-decomposition $\boldsymbol{B} = \boldsymbol{U}_B \boldsymbol{D}_B \boldsymbol{U}_B^T$ and $\boldsymbol{K} = \boldsymbol{U}_K \boldsymbol{D}_K \boldsymbol{U}_K^T$. Then we rewrite the two terms:

$$\begin{aligned} (\boldsymbol{B} \otimes \boldsymbol{K} + \sigma^2_{err}\boldsymbol{I})^{-1} &= (\boldsymbol{U}_B \boldsymbol{D}_B \boldsymbol{U}_B^T \otimes \boldsymbol{U}_K \boldsymbol{D}_K \boldsymbol{U}_K^T + \sigma^2_{err}\boldsymbol{I})^{-1} \\ &= (\boldsymbol{U}\boldsymbol{D}\boldsymbol{U}^T + \sigma^2_{err}\boldsymbol{I})^{-1} \\ &= \boldsymbol{U}(\boldsymbol{D} + \sigma^2_{err}\boldsymbol{I})^{-1}\boldsymbol{U}^T, \end{aligned}$$

where $\boldsymbol{U} = \boldsymbol{U}_B \otimes \boldsymbol{U}_K$ is a unitary matrix and $\boldsymbol{D} = \boldsymbol{D}_B \otimes \boldsymbol{D}_K$ is a diagonal matrix.

And

$$\begin{aligned}\log\det(\boldsymbol{B}\otimes\boldsymbol{K}+\sigma_{err}^2\boldsymbol{I}) &= \log\det(\boldsymbol{U}(\boldsymbol{D}+\sigma_{err}^2\boldsymbol{I})\boldsymbol{U}^T)\\ &= \log\det(\boldsymbol{D}+\sigma_{err}^2\boldsymbol{I})\,.\end{aligned}$$

Then applying Algorithm 14 in Saatçi (2012), the total computation cost is reduced to $O(\max(MN, M^3, N^3)) = O(\max(M^3, N^3))$.

In the general setting, model parameters $\sigma_{err}^2, \boldsymbol{L}, \tilde{\boldsymbol{\ell}}$ are of interest. Here $L_{ij}(t)$ is a three dimensional tensor and denote $\boldsymbol{L}_{ij} = (L_{ij}(t_1), \ldots, L_{ij}(t_n))^T$. The marginal posteriors of these parameters are

$$\begin{aligned}p(\sigma_{err}^2, \boldsymbol{L}, \tilde{\boldsymbol{\ell}}, \boldsymbol{\sigma}|\boldsymbol{y}, \boldsymbol{t}) &= \int p(\boldsymbol{f}, \sigma_{err}^2, \boldsymbol{L}, \boldsymbol{g}, \tilde{\boldsymbol{\ell}}, \boldsymbol{\sigma}|\boldsymbol{y}, \boldsymbol{t})d\boldsymbol{g}d\boldsymbol{f}\\ &\propto \mathcal{N}(\vec{\boldsymbol{y}}|\boldsymbol{0}, \boldsymbol{K}^f + \sigma_{err}^2 I)\prod_{i>j}\mathcal{N}(\boldsymbol{L}_{ij}|\boldsymbol{0}, K_L)\prod_i \mathcal{N}(\log \boldsymbol{L}_{ii}|\boldsymbol{0}, K_L)\\ &\quad \mathcal{N}(\tilde{\boldsymbol{\ell}}|\boldsymbol{0}, K_{\tilde{\boldsymbol{\ell}}})\mathrm{IG}(\sigma_{err}^2|a, b).\end{aligned} \tag{5.6}$$

We employ automatic gradient descent in pytorch to optimize all model parameters by maximizing the posterior in (5.5) or (5.6). Although our model is identifiable, the number of parameters is large, causing difficulty in optimization. Therefore, parameter initialization is critical, especially in (5.6).

We propose an empirical estimation approach for parameter initialization. We first set a window size w. Then given each time t_i, we assume our process $\boldsymbol{f}$ is locally smooth near any time t_i and so we assume that $\{\boldsymbol{y}(t_{i-w}), \ldots, \boldsymbol{y}(t_{i+w})\} \sim \mathcal{N}(\boldsymbol{0}, S(t_i))$. Then we first estimate $S(t_i)$ by $\hat{S}(t_i) = \frac{1}{2w}\sum_{j=i-w}^{i+w}\boldsymbol{y}_j\boldsymbol{y}_j^T$. Due to (5.3) and (5.4), we have $\mathrm{cov}(\boldsymbol{y}(t_i), \boldsymbol{y}(t_i)) = L(t_i)L(t_i)^T + \sigma_{err}^2\boldsymbol{I}$. As for the initialization, we estimate $L(t_i)$ by the cholesky decomposition of the MLE estimate of $\hat{S}(t_i)$ without considering variance of error σ_{err}^2. We estimate the the length-scale

parameters $\tilde{\ell}(t_i)$ by fitting the variogram of $\{y_m(t_{i-w}), \ldots, y_m(t_{i+w})\}$ in each dimension m and take the average of those parameter estimates. If any time index i is out of boundary, $[1, \ldots, N]$, we ignore those data in this procedure.

5.2.2 Model Prediction

For both separable and nonseparable models, given a new time stamp t^* with corresponding latent vector $\boldsymbol{f}^*$, we first consider the conditional distribution of $(\boldsymbol{y}, \boldsymbol{f}^*)$ given $\boldsymbol{\ell}, \ell^*, \boldsymbol{L}, \boldsymbol{L}^*, \sigma^2_{err}$, which is

$$\begin{pmatrix} \vec{\boldsymbol{y}} \\ \boldsymbol{f}^* \end{pmatrix} \Bigg| \boldsymbol{\ell}, \ell^*, \boldsymbol{L}, \boldsymbol{L}^*, \sigma^2_{err} \sim \mathcal{N}\left(\boldsymbol{0}, \begin{pmatrix} \boldsymbol{K}^f + \sigma^2_{err} I & \boldsymbol{k}^f \\ \boldsymbol{k}^{fT} & \boldsymbol{K}^{f*} \end{pmatrix}\right),$$

where $\boldsymbol{K}^f = K^f(\boldsymbol{t}, \boldsymbol{t})$, $\boldsymbol{K}^* = K^f(t^*, t^*)$ and $\boldsymbol{k}^{fT} = \text{cov}(\boldsymbol{f}^*, \vec{\boldsymbol{f}} + \boldsymbol{\epsilon}) = \text{cov}(\boldsymbol{f}^*, \vec{\boldsymbol{f}})$. Since $\text{cov}(\text{vec}[\boldsymbol{f}_1, \ldots, \boldsymbol{f}_N], \boldsymbol{f}^*) = A^f L(t^*)^T$, where

$$A^f = \begin{pmatrix} k(t^*, t_1)L(t_1) \\ k(t^*, t_2)L(t_2) \\ \vdots \\ k(t^*, t_N)L(t_N) \end{pmatrix},$$

it follows that

$$\begin{aligned} \boldsymbol{k}^f &= \text{cov}(\vec{\boldsymbol{f}}, \boldsymbol{f}^*) \\ &= \text{cov}(P(\text{vec}[\boldsymbol{f}_1, \ldots, \boldsymbol{f}_N]), \boldsymbol{f}^*) \\ &= P(\text{cov}(\text{vec}(\boldsymbol{f}), \boldsymbol{f}^*)), \end{aligned}$$

where P is a permutation operator such that $P(\text{vec}([\boldsymbol{f}_1, \ldots, \boldsymbol{f}_N])) = \text{vec}([\boldsymbol{f}_1, \ldots, \boldsymbol{f}_N]^T)$. The conditional distribution of f^* is indeed a multivariate Gaussian distribution

such that

$$p(\boldsymbol{f}^*|\boldsymbol{y}, \boldsymbol{\ell}, \ell^*, \boldsymbol{L}, \boldsymbol{L}^*) = \mathcal{N}(\boldsymbol{f}^*|\boldsymbol{\mu}^*, \Sigma^*),$$

where the conditional mean is $\boldsymbol{\mu}^* = \boldsymbol{k}^{fT}(\boldsymbol{K}^f + \sigma^2_{err}\boldsymbol{I})^{-1}\vec{\boldsymbol{y}}$ and conditional covariance matrix is $\Sigma^* = \boldsymbol{K}^{f*} - \boldsymbol{k}^{fT}(\boldsymbol{K}^f + \sigma^2_{err}\boldsymbol{I})^{-1}\boldsymbol{k}^f$.

On the other hand, the posterior predictive distribution of * can be approximated with MAP estimates:

$$p(\boldsymbol{f}^*|\boldsymbol{y}) = \int p(\boldsymbol{f}^*|\boldsymbol{y}, \boldsymbol{\ell}, \ell^*, \boldsymbol{L}, \boldsymbol{L}^*)p(\ell^*|\boldsymbol{\ell})p(L^*|L)p(\boldsymbol{\ell}, \boldsymbol{L}, \sigma^2_{err}|\boldsymbol{y})d(\ell^*, L^*, \boldsymbol{\ell}, \boldsymbol{L}, \sigma^2_{err}) \tag{5.7}$$

$$\approx \int \mathcal{N}(\boldsymbol{f}^*|\hat{\boldsymbol{\mu}}^*, \hat{\Sigma}^*)p(\ell^*|\hat{\boldsymbol{\ell}})p(\boldsymbol{L}^*|\hat{\boldsymbol{L}})d(\ell^*, \boldsymbol{L}^*), \tag{5.8}$$

where $\hat{\boldsymbol{\mu}}^*, \hat{\Sigma}^*$ are estimates of $\boldsymbol{\mu}^*$ and Σ^* by simply replacing $\boldsymbol{\ell}, L, \sigma^2_{err}$ with their MAP estimates. For fully Baysian inference, the posterior predictive of $\boldsymbol{f}^*$ can be sampled using (5.7) while for the MAP inference the posterior predictive of $\boldsymbol{f}^*$ can be approximated by sampling under (5.8).

Moreover, the posterior correlation matrix of underlying multivariate process $\boldsymbol{f}$ at time t^* is approximated via MAP estimates as

$$\begin{aligned} p(\boldsymbol{R}^*|\boldsymbol{y}) &= p(\mathrm{diag}(K^*)^{-\frac{1}{2}}\boldsymbol{K}^*\mathrm{diag}(K^*)^{-\frac{1}{2}}|\boldsymbol{y}) \\ &= \int p(\mathrm{diag}(\boldsymbol{K}^*)^{-\frac{1}{2}}\boldsymbol{K}^*\mathrm{diag}(\boldsymbol{K}^*)^{-\frac{1}{2}}|\boldsymbol{L})p(\boldsymbol{L}|\boldsymbol{y})d\boldsymbol{L} \\ &\approx p(\mathrm{diag}(\boldsymbol{K}^*)^{-\frac{1}{2}}\boldsymbol{K}^*\mathrm{diag}(\boldsymbol{K}^*)|\hat{\boldsymbol{L}}. \end{aligned} \tag{5.9}$$

5.2.3 Model Evaluation

We assess model performance by simulating independent replicates from the posterior prediction distribution at each observed time. It suggests that for each observed time, t_i, we sample replicates $\boldsymbol{y}(t_{rep,i})$ given observations $\boldsymbol{y}$, based on the posterior predictive sampling in (5.7) or (5.8). Then we summarize them using both posterior predictive mean $\boldsymbol{\mu}_{rep,i}$ and variance $\Sigma_{rep,i}$. We prefer to use a loss function which penalizes both departure of predictive means from their observed values and excessive uncertainty in the predictive data. Therefore, we use a squared error loss function in Gelfand and Ghosh (1998), where the measures for these two criteria are evaluated as $G = \sum_{i=1}^{N} \|\boldsymbol{y}_i - \boldsymbol{\mu}_{rep,i}\|^2$, where $\|\cdot\|$ is the standard Euclidean norm and $P = \sum_{i=1}^{N} \mathrm{tr}(\Sigma_{rep,i})$. The model selection criteria is the score $D = G + P$. The lower score indicates the better model. Original G, P and D are used for model comparison to measure the performance on model fitting. To measure the model prediction performance, we use the predictive mean square error (PMSE) $\sum_{n=1}^{N^*} \|\boldsymbol{y}_i^* - \boldsymbol{\mu}_i^*\|^2$ where we estimate each predictor using the corresponding posterior predictive mean $\boldsymbol{\mu}_i^*$.

5.3 Experiments

We validate our proposed models on synthetic data and Kaiser Permanente's Electronic Health Records (EHR) data (Fohner et al., 2019). all models/inference algorithms are implemented in Python with the open source Pytorch library.

5.3.1 Synthetic Data

In this section, we first show that our inference is capable of recovering latent processes. Then we illustrate that our proposed models including GNMGP and

SNMGP have better model fitting and model prediction performance than a naive LMC. We also discuss which cases our SNMGP is preferred in.

We first uniformly generate 200 timestamps $\boldsymbol{t} = (t_1, \ldots, t_{200})^T$ on a unit interval $(0, 1)$. Then we generate a bi-variate time series $\boldsymbol{y} = (\boldsymbol{y}_1, \ldots, \boldsymbol{y}_{200})^T$ from the GNMGP model, in which we set deterministic latent processes. They include a log length-scale process $l(t) = 8(t-1)^3$, two standard deviation processes $s_1(t) = 1+t^2$ and $s_2(t) = 2-t^2$, and a correlation process $r(t)$ between two dimensions $r(t) = \cos(\pi t)$. We assume a small standard deviation of error $\sigma_{err} = 0.001$. Those latent processes are shown in Figure 5.1 as red dashed lines. Based on the same timestamps $\boldsymbol{t}$ and the same latent processes, we repeatedly generate 100 samples, denoted as $\{\boldsymbol{y}^{(r)}\}_{r=1}^{100}$.

To show our inference is capable to recovery true latent processes, for each sample $\boldsymbol{y}^{(r)}$, we inference latent processes via our proposed MAP method.

Specifically we select an informative prior for variance of measurement error such that $\sigma_{err}^2 \sim \text{IG}(10^{-6}, 10^{-6})$ and a prior of length-scale process $\ell \sim \text{logGP}(0, \text{RBF}(\sigma = 10, \text{d} = 1))$. We set the prior of L as $L_{ij}(t) \sim \text{GP}(0, \text{RBF}(\sigma = 1, d = 1))$ for $i \neq j$ and $L_{ii}(t) \sim \text{logGP}(0, \text{RBF}(\sigma = 1, d = 1))$. In addition, we employ our propose empirical estimation approach for initialization. After obtaining estimated latent processes for all 100 samples , we carry out functional Box-plot (Sun and Genton, 2011; Meng et al., 2017) to summarize those estimated processes in Figure 5.1, comparing with their true latent processes. The dark region corresponds to 50% central region while the grey region refers to the functional data envelope. The results show that our inference can recover the true latent processes based on 50% central area and the functional median. We also note that those estimates depend on their empirical estimates because the marginal posterior 5.6 is non-convex.

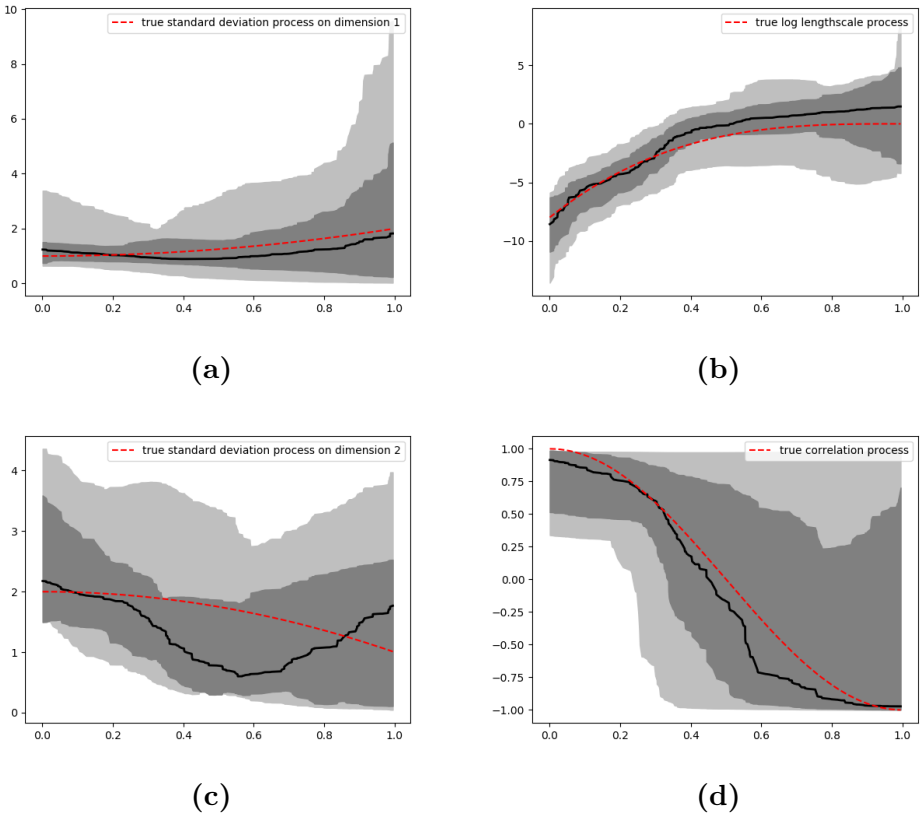

Figure 5.1: (5.1a) 95% functional boxplot for estimated standard deviation process on dimension 1, (5.1b): log length-scale processes, (5.1d): correlation process across dimensions 1 and 2.

After our proposed inference is illustrated to be capable of recovering latent processes, we are going to evaluate model fitting and model prediction performance of our proposed models GNMGP and SNMGP and compare it with a naive LMC model.

To make three models comparable, we would tune different priors on each model and select the one which leads to best prediction results. We note that prior selection is important in our model to avoid overfitting issue.

SNMGP and GNMGP models are initialized based on our proposed empirical estimation approach. In this experiment, we use the same synthetic data from

last experiment, but for each sample, we randomly take 33% observations for testing and take the rest 66% observations for training. We train all models for a maximum of 2000 epochs (training iterations) using Adam (Kingma and Ba, 2014) with a learning rate of 0.01.

The performance of model fitting is summarized by evaluation metrics G, P and D we discussed in Section 5.2.3 and the performance of model prediction is summarized by predictive mean square error (PMSE).

We summarize those metrics of all samples by their mean and standard deviation in Table 5.1. We also summarize the total running time for different models on cluster using 100 cores in parallel.

The model evaluation results illustrate that our SNMGP and GNMGP outperform LMC significantly especially for model fitting. This is because the naive LMC mis-specify the time varying smoothness and leads to large estimated measurement error while our proposed SNMGP and GNMGP can capture the correct scale of measurement error. The running time suggests that the inference of SNGMP is significantly faster than that of GNMGP, even though GNMGP performance is better than SNMGP.

	LMC	SNMGP	GNMGP
G	101.95(63.47)	10.17(31.12)	6.26(22.01)
P	121.08(63.13)	12.73(37.58))	7.33(25.11)
D	223.02(125.84)	22.89(68.60)	13.60(47.11)
PMSE	0.94(0.57)	0.76(1.27)	0.56(0.39)
Total running time	2min	9 min	57 min

Table 5.1: Evaluation metrics of model fitting and prediction and running time on LMC, SNMGP and GNMGP for 100 replications of synthetic data generated from GNMGP.

Table 5.1 indicates that the model prediction performance of SNMGP is weak, since the standard deviation of PMSE is significantly larger than both LMC and

GNMGP. Therefore, we would study the case in which the SNMGP should be significantly better than LMC.

We state that SNMGP have a more robust prediction performance when $L(t)$ is allowed to decompose as $L(t) = Ls(t)$ and justify it via a new experiment. In this experiment, we generate 100 two-dimensional multivariate time series from SNMGP model itself, where the correlation between dimensions is time invariant. Particularly, in the SNMGP, we define $B = LL^T = \begin{pmatrix} 1 & 1 \\ 1 & 4 \end{pmatrix}$, $s(t) = 1 + t^2$, which suggests that the correlation between two dimension is fixed as 0.5. We also define the fixed length-scale process $y = 8(t-1)^3$. And in each sample we randomly take 33% data for testing and 66% for training. We summarize modeling fitting by evaluation metrics G, P and D and model prediction by PMSE in Table 5.2.

	LMC	SNMGP
G	109.44(76.32)	11.75(30.11)
D	122.60(72.72)	15.26(38.55)
P	232.05(146.09)	27.00(68.56)
PMSE	1.06(0.57)	0.64(0.43)

Table 5.2: Evaluation metrics of model prediction on LMC and SNMGP on 100 replications of synthetic data generated from SNMGP.

5.3.2 Kaiser Permanente Electronic Health Records Data

We demonstrate our proposed framework by modeling the time-varying vital signs such as, systolic blood pressure (BPSYS), diastolic blood pressure (BPDIA), pulse pressure (PP), heart rate (HRTRT) and oxygen saturation (O2SAT) of patients admitted to the emergency department (ED) with confirmed or suspected infection. The Kaiser Permanente (KP) dataset is an anonymized EHR dataset where a patient's hospital stay is identified by an episode ID (Fohner et al., 2019; Seymour et al., 2016).

We randomly select one patient who has sepsis for illustration. We extract vitals BPSYS, BPDIA, PP, HRTRT and O2SAT and remove the missing data. Then we get multivariate time series with 138 time stamps. We take 96 time stamps for training and take the reminder 42 time stamps for testing. We first standardized data to zero mean and one standard deviation for each vital individually.

The priors of LMC, SNMGP and GNMGP are discussed as follows. We set an non-informative prior for the variance of measurement error such that $\sigma^2_{err} \sim \text{IG}(1,1)$ for all three models. As for the prior of length-scale parameter ℓ, we set $\ell \sim \text{logN}(0,1)$, where logN refers to log Normal distribution and set $\ell \sim \text{logGP}(0, \text{RBF}(\sigma = 10, d = 1))$ for SNMGP and GNMGP. With respect to the prior of L, we set $L_{ij} \sim \text{N}(0,1)$ for LMC, set $L_{ij} \sim \text{N}(0,1)$ for $i \neq j$, $L_{ii} \sim \text{logN}(0,1)$ and $s(t) \sim \text{logGP}(0, \text{RBF}(\sigma = 1, d = 1))$ for SNMGP and set $L_{ij} \sim \text{GP}(0, \text{RBF}(\sigma = 1, d = 1))$ for $i \neq j$ and $L_{ii} \sim \text{logGP}(0, \text{RBF}(\sigma = 1, d = 1))$.

During the inference, the initialization is based on proposed empirical estimation. However, for SNMGP and GNMGP, tuning the starting points for the length-scale parameters is required since variogram fitting is not robust in the real data. We do inference with initialization in which we initialize length scale parameters with different constants and then take the optimal solution.

The model selection is based on G, P, D evaluation metrics shown in Table 5.3. After we obtained samples based on standardized data, we convert them back to the original scale and compute the root mean square error (RMSE) and log predictive density (LPD). Mathematically, assume that we have predictive time $\boldsymbol{t}^* = (t^*_1, \ldots, t^*_{N^*})$. We denote original observations $\tilde{\boldsymbol{y}}^*_i = (\tilde{y}^*_{i,1}, \ldots, \tilde{y}^*_{i,M})^T$ at time t^*_i and we have posterior predictive mean $\tilde{\boldsymbol{\mu}}^*_i = (\tilde{\mu}^*_{i,1}, \ldots, \tilde{\mu}^*_{i,M})^T$ and posterior predictive standard deviation $\tilde{\boldsymbol{\sigma}}^*_i = (\tilde{\mu}^*_{i,1}, \ldots, \tilde{\mu}^*_{i,M})^T$ on the original scale at time t^*_i. Then root mean square error is defined as $\text{RMSE} = \sqrt{\frac{1}{N^*} \sum_{i=1}^{N^*} \|\tilde{\boldsymbol{y}}^*_i - \tilde{\boldsymbol{\mu}}^*_i\|^2}$ and

log predictive density is defined as $\sum_{i=1}^{N^*}\sum_{j=1}^{M}\mathcal{N}(\tilde{y}^*_{i,j}|\tilde{\mu}^*_{i,j},\tilde{\sigma}^{*2}_{i,j})$. Those results are summarized in Table 5.3.

	MAP			HMC		
	LMC	SNMGP	GNMGP	LMC	SNMGP	GNMGP
G	172.27	169.50	156.19	176.93	168.70	161.18
P	264.47	244.90	231.31	280.48	245.81	271.44
D	436.74	414.40	387.51	457.41	414.51	432.62
PMSE	123.72	118.73	111.32	121.02	116.71	110.30
RMSE	7.44	7.18	7.10	7.38	7.12	7.00
LPD	-3.23	-3.24	-3.19	-3.23	-3.21	-3.19

Table 5.3: Evaluation metrics of model fitting and assessment for three models for one patient's records from KAISER under both standardized scale and original scale.

Moreover, in order to show the population's characteristics, we considered two cohorts. We randomly sampled 1000 patients with sepsis and 1000 patients with nonsepsis whose number of records is in the interval $(50, 500)$, based on the empirical age density of the whole population for each cohort. Then we normalized each records via each dimension and fit a LMC model. We initialize length scale parameters using an constant function at the length-scale estimate in the LMC model and we initialize the other parameters based on the empirical estimation approach. Then we carry out MAP inference for all patients in parallel and sample 100 times from posterior predictive distributions of lengthscale parameters $\boldsymbol{\ell}^*$ and correlation parameters $\boldsymbol{R}^*$ defined in (5.9) at the same observed time on LAPS2, $\boldsymbol{t}^* = (t^*_1, \ldots, t^*_{N^*})$, which is an equally-space hourly sequence since the patient comes to ICU.

Motivated from Fairchild et al. (2016) such that exploration of the cross-correlation of different vitals would lead to earlier treatment, we define the correlation coefficient indicator CCI(ψ) for interested parameter ψ such as the lengthscale parameter or any cross correlation coefficient parameter. CCI(ψ) is defined via

the empirical Pearson correlation coefficient. Letting $\boldsymbol{\psi} = (\psi_1, \ldots, \psi_{N^*})^T$, where $\psi_i = \psi(t_i^*)$ and LAPS2 observations $\text{LAPS2} = (\text{LAPS2}_1, \ldots, \text{LAPS2}_{N^*})$, where $\text{LAPS2}_i = \text{LAPS2}(t_i^*)$. Then

$$\text{CCI}(\psi) = \frac{\sum_{i=1}^{N^*}(\psi_i - \overline{\psi})(\text{LAPS2}_i - \overline{\text{LAPS2}})}{\sqrt{\sum_{i=1}^{N^*}(\psi_i - \overline{\psi})^2}\sqrt{\sum_{i=1}^{N^*}(\text{LAPS2}_i - \overline{\text{LAPS2}})^2}} \tag{5.10}$$

Due to the fact that LAPS2 is a proposed health indicator by Kaiser Permanente, our summary statistics CCI is used to illustrate the correlation between smoothness or cross correlation between any pair of vitals and patient health status.

Based on the posterior samples, we sample the posterior predictive correlation coefficient indicator $\text{CCI}(\psi)$ between LAPS2 and length-scale parameter or any pair-wise correlation parameter. For each patient, we compute the 95% credible interval of those posterior predictive CCI and summarize them across patients by compare the credible intervals $\text{CCICI}(\psi)$ with 0. we classify them as negative, natural and positive relation if CCICI is on the left, include or on the right of 0. The summarized results for two cohorts are shown in Table 5.4 and Table 5.5.

Both tables show a consist result such that a majority of patients have a negative correlation between their health condition and smoothness of records. It indicates that when patients have a non smooth records, it is more likely that patients are in a bad health condition. Moreover the cross correlation between BPDIA and BPSYS has a positive correlation with the health condition in a majority of patients. It probably means that as for healthy patients, it is more likely that their BPDIA and BPSYS have a positive relation. Comparing the results in sepsis group with those in nonsepsis group, unfortunately there are only slight difference for those relations and thus it may not be clinical useful to distinguish sepsis from nonsepsis based on our model.

	Sepsis			Nonepsis		
Feature	Negative	Neutral	Positive	Negative	Neutral	Positive
Length scale	0.712	0.005	0.283	0.677	0.002	0.321
BPDIA vs BPSYS	0.399	0.005	0.596	0.389	0.004	0.607
BPDIA vs HRTRT	0.455	0.01	0.535	0.441	0.008	0.551
BPDIA vs O2SAT	0.501	0.009	0.490	0.521	0.006	0.473
BPDIA vs PP	0.476	0.004	0.52	0.482	0.01	0.508
BPSYS vs HRTRT	0.45	0.003	0.547	0.431	0.006	0.563
BPSYS vs O2SAT	0.53	0.007	0.463	0.51	0.007	0.483
BPSYS vs PP	0.554	0.007	0.439	0.575	0.007	0.418
HRTRT vs O2SAT	0.526	0.004	0.47	0.52	0.01	0.47
HRTRT vs PP	0.439	0.009	0.552	0.457	0.007	0.536
O2SAT vs PP	0.509	0.011	0.48	0.514	0.002	0.484

Table 5.4: Coverage rate for different features in Cohort A

	Sepsis			Nonepsis		
Feature	Negative	Neutral	Positive	Negative	Neutral	Positive
Length scale	0.715	0.004	0.281	0.655	0.003	0.342
BPDIA vs BPSYS	0.418	0.004	0.578	0.38	0.004	0.616
BPDIA vs HRTRT	0.449	0.011	0.54	0.468	0.006	0.526
BPDIA vs O2SAT	0.504	0.007	0.489	0.514	0.006	0.48
BPDIA vs PP	0.484	0.007	0.509	0.466	0.008	0.526
BPSYS vs HRTRT	0.43	0.004	0.566	0.476	0.004	0.529
BPSYS vs O2SAT	0.53	0.005	0.465	0.527	0.008	0.465
BPSYS vs PP	0.555	0.005	0.44	0.546	0.008	0.446
HRTRT vs O2SAT	0.53	0.004	0.466	0.527	0.008	0.465
HRTRT vs PP	0.462	0.004	0.534	0.487	0.003	0.51
O2SAT vs PP	0.511	0.007	0.482	0.48	0.001	0.519

Table 5.5: Coverage rate for different features in Cohort B

Chapter 6

Discussion

6.1 Summary of Contributions

The main contribution of this dissertation is to develop interpretable and flexible models for temporal data using latent processes such as Markov jump processes and Gaussian processes. This is elaborated in several parts: First, Chapter 2 proposes a hierarchical model for cervical cancer screening test data, in which heterogeneous time continuous Markov jump processes are considered to model the latent dynamical system of patients' states. We propose a finite mixture structure to model the individual heterogeneity and we also incorporate both treatment information and censored information in our model through graphical structure. Next, Chapter 3 presents a systematic overview of sparse Gaussian processes and give a general regularization framework for inducing-point based sparse Gaussian processes and extend it to latent variable models. Particularly, we explore the variation inference under our regularization framework and we theoretically prove that the objective function under our regularization framework is a variational lower bound in a corresponding empirical Bayesian model. Our regularization framework is illustrated on different methods, different hyper-parameter settings

and different datasets. Based on our regularization framework, we propose a hierarchical sparse latent Gaussian processes to model temporal categorical data as well as corresponding efficient variational inference. Our model is illustrated on both synthetic data and Stock index data. Finally, Chapter 5 proposes a flexible class of nonstationary multivariate Gaussian process models which is allowable to model both time-varying smoothness, time-varying scale and time-varying correlation. Particularly, we focus on the analysis of electronic health records data to understand the latent correlation across different clinical variables across time. Our inference is based on both Hamiltonian Monte Carlo as a fully Bayesian inference and Maximum a posteriori as a approximate Bayesian inference due to its computational benefits. Our model is illustrated on both synthetic data and electronic health records data from Kaiser Permanente.

6.2 Future Work

Future work for the hidden Markov model can be the use of deep learning methods as a surrogate model for the computation of the matrix exponential, which is the most expensive part of the current model. Another potential work related to this project is to model the recurrence event. One possible way to deal with it is to add recurrence effect on the intensity functions after a cancer treatment.

One potential next direction, which I am working on is motivated by the nonstationary multivariate Gaussian processes (NMGP). Directly modeling the latent processes in the NMGP via Gaussian processes is expensive especially for massive data and high dimensional data. It is because that the number of parameters in the model is proportional to the number of observations. In the extension work, instead of using Gaussian processes to model latent processes, we model

those latent processes via B splines, in which we have locally compact functional bases. The number of interested parameters is only related to the number of functional bases rather than the number of observations. We would discuss the difference between our model and other models including process convolution and Karhunen-Loeve expansion.

Another potential direction is to introduce the sparse Gaussian process and doubly stochastic variational inference into the nonstationary multivariate Gaussian processes to alleviate the computational burden.

6.3 Conclusion

In the dissertation, we discuss different approaches to model temporal data via different latent processes. As for the computation for large datasets, we propose and discuss different efficient inference methods including EM algorithm, Hamiltonian Monte Carlo and variational inference approaches. I hope this dissertation will contribute to our understanding of the strengths and weaknesses of these different approaches and contribute methodologies to temporal data modeling especially in the healthcare research field.

Appendix A

Appendix

A.1 Advanced MCMC

This section, we introduce three advanced MCMC algorithms. First of all, we introduce a traditional Metropolis Hasting framework.

Sampling $\boldsymbol{\theta} \sim p(\boldsymbol{\theta})$ is of interest. We set the proposal transition distribution as $q(\boldsymbol{\theta}', \boldsymbol{\theta})$. Then the recursive procedures are displayed as

- Sample new parameters $\boldsymbol{\theta}_{new} \sim q(\boldsymbol{\theta}_{new}|\boldsymbol{\theta}^{(i)})$.
- Compute the accept rate $r = \min\left(\frac{p(\boldsymbol{\theta}_{new})q(\boldsymbol{\theta}|\boldsymbol{\theta}_{new})}{p(\boldsymbol{\theta})q(\boldsymbol{\theta}_{new}|\boldsymbol{\theta})}, 1\right)$.
- Accept $\boldsymbol{\theta}^{(i+1)} = \boldsymbol{\theta}_{new}$ with probability r, otherwise, keep previous parameters $\boldsymbol{\theta}^{(i+1)} = \boldsymbol{\theta}^{(i)}$.

Three algorithms are based on this Metropolis Hasting framework and they are discussed as follows.

A.1.1 Langevin Adapted Metropolis Hasting

To utilize the gradient information, the proposal structure in Besag et al. (1995) is

$$\boldsymbol{\theta}_{new} = \boldsymbol{\theta} + \frac{\epsilon^2}{2} g(\boldsymbol{\theta}) + \epsilon \mathcal{N}(\mathbf{0}, \boldsymbol{I})$$

where $g(\boldsymbol{\theta}) = \nabla \log p(\boldsymbol{\theta})$.

It suggests the proposal kernel is

$$q(\boldsymbol{\theta}_{new}, \boldsymbol{\theta}) = \exp\left[-\frac{1}{2\epsilon^2} \| \boldsymbol{\theta} - \boldsymbol{\theta}_{new} - \frac{\epsilon^2}{2} g(\boldsymbol{\theta}_{new}) \|^2 \right].$$

A.1.2 Hessian Adapted Metropolis Hasting

Qi and Minka (2002) propose the Hessian adaptive Metropolis Hasting approach which utilizes both the gradient and the Hessian matrix information in its proposal structure

$$\boldsymbol{\theta}_{new} = \boldsymbol{\theta} + \epsilon \Sigma(\boldsymbol{\theta}) g(\boldsymbol{\theta}) + \mathcal{N}(\mathbf{0}, \Sigma(\boldsymbol{\theta})) \tag{A.1}$$

where $\Sigma(\boldsymbol{\theta}) = -\left(\nabla \nabla^T \log p(\boldsymbol{\theta}) \right)^{-1}$.

It suggests the proposal kernel is

$$q(\boldsymbol{\theta}_{new}, \boldsymbol{\theta}) = \mathcal{N}(\boldsymbol{\theta}_{new} - \boldsymbol{\theta} + \epsilon \Sigma(\boldsymbol{\theta}) g(\boldsymbol{\theta}), \Sigma(\boldsymbol{\theta})) \tag{A.2}$$

A.1.3 Fisher Adapted Metropolis Hasting

One concern of the Hessian adapted Metropolis Hasting approach is the matrix inverse problem. Numerically the Hessian matrix may not be positive definite.

The idea of replacing Hessian matrix by the negative fisher information matrix has been explored. We proposed a simple proposal structure

$$\boldsymbol{\theta}_{new} = \boldsymbol{\theta} + \mathcal{N}(\mathbf{0}, \Sigma^*)$$

where $\Sigma^* = -(\nabla\nabla^T(\boldsymbol{\theta}^*))^{-1}$ is the inverse of the observed fisher information matrix and $\boldsymbol{\theta}^*$ is the maximum likelihood estimates for $\log p(\boldsymbol{\theta})$. Then it is a symmetric random walk with proposal distribution such that

$$q(\boldsymbol{\theta}_{new}, \boldsymbol{\theta}) = \mathcal{N}(\boldsymbol{\theta}_{new} - \boldsymbol{\theta}, \Sigma^*)$$

where $q(\boldsymbol{\theta}_{new}|\boldsymbol{\theta})$ and $q(\boldsymbol{\theta}|\boldsymbol{\theta}_{new})$ cancel out in the computation of accept rate. The computation of Fisher Adapted Metropolis Hasting is much cheaper than Hessian adapted Metropolis Hasting since it does not need to compute the proposal transition kernel in each iteration.

A.2 Matrix Gradient

Let $a_{ij} \in \mathbb{R}, i = 1, \ldots, m, j = 1, \ldots, n$. Then the real matrix $\boldsymbol{A}$ is expressed as

$$\boldsymbol{A} = \begin{pmatrix} a_{11} & a_{12} & \ldots & a_{1n} \\ a_{21} & a_{22} & \ldots & a_{2n} \\ \vdots & \vdots & & \vdots \\ a_{m1} & a_{m2} & \ldots & a_{mn} \end{pmatrix}.$$

Assume f is a mapping from $\mathbb{R}^{m\times n}$ to $\mathbb{R}$. Then the gradient of $f(\boldsymbol{A})$ can be expressed as

$$\frac{\partial}{\partial \boldsymbol{A}} f(\boldsymbol{A}) = \boldsymbol{A}' = \begin{pmatrix} \frac{\partial}{\partial a_{11}} & \frac{\partial}{\partial a_{12}} & \cdots & \frac{\partial}{\partial a_{1n}} \\ \frac{\partial}{\partial a_{21}} & \frac{\partial}{\partial a_{22}} & \cdots & \frac{\partial}{\partial a_{2n}} \\ \vdots & \vdots & & \vdots \\ \frac{\partial}{\partial a_{m1}} & \frac{\partial}{\partial a_{m2}} & \cdots & \frac{\partial}{\partial a_{mn}} \end{pmatrix} \boldsymbol{A}\,.$$

Bibliography

Alaa, A. M. and van der Schaar, M. (2017). Bayesian Inference of Individualized Treatment Effects using Multi-task Gaussian Processes. In Guyon, I., Luxburg, U. V., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S., and Garnett, R., editors, *Advances in Neural Information Processing Systems 30*, pages 3424–3432. Curran Associates, Inc.

Alaa, A. M., Yoon, J., Hu, S., and Schaar, M. v. d. (2018). Personalized Risk Scoring for Critical Care Prognosis Using Mixtures of Gaussian Processes. *IEEE Transactions on Biomedical Engineering*, 65(1):207–218.

Alvarez, M. A., Rosasco, L., Lawrence, N. D., et al. (2012). Kernels for vector-valued functions: A review. *Foundations and Trends® in Machine Learning*, 4(3):195–266.

Anderes, E. B. and Stein, M. L. (2011). Local likelihood estimation for nonstationary random fields. *Journal of Multivariate Analysis*, 102(3):506–520.

Apanasovich, T. V. and Genton, M. G. (2010). Cross-covariance functions for multivariate random fields based on latent dimensions. *Biometrika*, 97(1):15–30.

Backurs, A. and Tzamos, C. (2017). Improving viterbi is hard: Better runtimes imply faster clique algorithms. pages 311–321.

Banerjee, S., Gelfand, A. E., Finley, A. O., and Sang, H. (2008). Gaussian predictive process models for large spatial data sets.

Bao, Y., Kuang, Z., Peissig, P., Page, D., and Willett, R. (2017). Hawkes process modeling of adverse drug reactions with longitudinal observational data. In Doshi-Velez, F., Fackler, J., Kale, D., Ranganath, R., Wallace, B., and Wiens, J., editors, *Proceedings of the 2nd Machine Learning for Healthcare Conference*, volume 68 of *Proceedings of Machine Learning Research*, pages 177–190, Boston, Massachusetts. PMLR.

Baydin, A. G., Pearlmutter, B., Andreyevich Radul, A., and Siskind, J. (2018a). Automatic differentiation in machine learning: A survey. *Journal of Machine Learning Research*, 18:1–43.

Baydin, A. G., Pearlmutter, B. A., Radul, A. A., and Siskind, J. M. (2018b). Automatic differentiation in machine learning: a survey. *Journal of machine learning research*, 18(153).

Berchtold, A. and Raftery, A. (2002). The mixture transition distribution model for high-order markov chains and non-gaussian time series. *Statist. Sci.*, 17(3):328–356.

Bernhard, S., Alexander, S., and Klaus-Robert, M. (1998). Kernel principle component analysis. *Advances in Kernel Methods - Support Vector Learning*, pages 327–352.

Besag, J., Green, P., Higdon, D., and Mengersen, K. (1995). Bayesian computation and stochastic systems. *Statist. Sci.*, 10(1):3–41.

Bhat, K. S., Haran, M., Goes, M., and Chen, M. (2010). Computer model calibration with multivariate spatial output: A case study. *Frontiers of Statistical Decision Making and Bayesian Analysis*, pages 168–184.

Bishop, C. M. (2006). *Pattern recognition and machine learning*. Springer Science+ Business Media.

Bladt, M. and Sørensen, M. (2005). Statistical inference for discretely observed markov jump processes. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 67(3):395–410.

Bourgault, G. and Marcotte, D. (1991). Multivariable variogram and its application to the linear model of coregionalization. *Mathematical Geology*, 23(7):899–928.

Bowman, S. R., Vilnis, L., Vinyals, O., Dai, A. M., Józefowicz, R., and Bengio, S. (2016). Generating sentences from a continuous space.

Bray, F., Ferlay, J., Soerjomataram, I., Siegel, R. L., Torre, L. A., and Jemal, A. (2018). Global cancer statistics 2018: GLOBOCAN estimates of incidence and mortality worldwide for 36 cancers in 185 countries. *CA: A Cancer Journal for Clinicians*, 68(6):394–424.

Brooks, S., Gelman, A., Jones, G., and Meng, X.-L. (2011). *Handbook of markov chain monte carlo*. CRC press.

Brunsdon, C., Fotheringham, A. S., and Charlton, M. E. (1996). Geographically weighted regression: a method for exploring spatial nonstationarity. *Geographical analysis*, 28(4):281–298.

Bureau, A., Shiboski, S., and Hughes, J. P. (2003). Applications of continuous time hidden markov models to the study of misclassified disease outcomes. *Statistics in Medicine*, 22(3):441–462.

Bush, C. and Macechern, S. (1996). A semiparametric bayesian model for randomised block designs. *Biometrika*, 83(2):275–285.

Canfell, K., Barnabas, R., Julietta, P., and Valerie, B. (2004). The predicted effect of changes in cervical screening practice in the uk: results from a modelling study. *British journal of cancer*, 91(3):530–536.

Cao, H., Lake, D. E., Griffin, M. P., and Moorman, J. R. (2004). Increased Nonstationarity of Neonatal Heart Rate Before the Clinical Diagnosis of Sepsis. *Annals of Biomedical Engineering*, 32(2):233–244.

Cheng, L.-F., Darnell, G., Dumitrascu, B., Chivers, C., Draugelis, M. E., Li, K., and Engelhardt, B. E. (2017). Sparse Multi-Output Gaussian Processes for Medical Time Series Prediction. *arXiv e-prints*, page arXiv:1703.09112.

Cho, K., van Merrienboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., and Bengio, Y. (2014). Learning Phrase Representations using RNN Encoder-Decoder for Statistical Machine Translation. *arXiv e-prints*, page arXiv:1406.1078.

Choi, E., Bahadori, M. T., Schuetz, A., Stewart, W. F., and Sun, J. (2016). Doctor ai: Predicting clinical events via recurrent neural networks. In *Machine Learning for Healthcare Conference*, pages 301–318.

Chung, J., Gulcehre, C., Cho, K., and Bengio, Y. (2014). Empirical evaluation of gated recurrent neural networks on sequence modeling.

Connor, R. J. and Mosimann, J. E. (1969). Concepts of independence for proportions with a generalization of the dirichlet distribution. *Journal of the American Statistical Association*, 64(325):194–206.

Cressie, N. and Wikle, C. K. (2011). *Statistics for Spatio-Temporal Data*. Wiley.

Damianou, A. C., Titsias, M. K., and Lawrence, N. D. (2016). Variational inference for latent variables and uncertain inputs in gaussian processes. *Journal of Machine Learning Research*, 17(42):1–62.

Das, S. and Nason, G. P. (2016). Measuring the degree of non-stationarity of a time series. *Stat*, 5(1):295–305.

Doerr, C., Blenn, N., and Van Mieghem, P. (2013). Lognormal infection times of online information spread. *PloS one*, 8(5).

Doetsch, P., Kozielski, M., and Ney, H. (2014). Fast and robust training of recurrent neural networks for offline handwriting recognition. In *2014 14th International Conference on Frontiers in Handwriting Recognition*, pages 279–284.

Duchi, J., Hazan, E., and Singer, Y. (2010). Adaptive subgradient methods for online learning and stochastic optimization. Technical Report UCB/EECS-2010-24, EECS Department, University of California, Berkeley.

Dürichen, R., Pimentel, M. A. F., Clifton, L., Schweikard, A., and Clifton, D. A. (2014). Multi-task Gaussian process models for biomedical applications. In *IEEE-EMBS International Conference on Biomedical and Health Informatics (BHI)*, pages 492–495.

Dyer, C., Ballesteros, M., Ling, W., Matthews, A., and Smith, N. A. (2015). Transition-Based Dependency Parsing with Stack Long Short-Term Memory. *arXiv e-prints*, page arXiv:1505.08075.

Eidsvik, J., Finley, A. O., Banerjee, S., and Rue, H. (2012). Approximate bayesian inference for large spatial datasets using predictive process models. *Computational Statistics & Data Analysis*, 56(6):1362 – 1380.

Ek, C. H., Torr, P. H., and Lawrence, N. D. (2007). Gaussian process latent variable models for human pose estimation. In *International workshop on machine learning for multimodal interaction*, pages 132–143. Springer.

Escobar, G. J., Gardner, M. N., Greene, J. D., Draper, D., and Kipnis, P. (2013). Risk-adjusting hospital mortality using a comprehensive electronic record in an integrated health care delivery system. *Medical care*, pages 446–453.

Fairchild, K. D., Lake, D. E., Kattwinkel, J., Moorman, J. R., Bateman, D. A., Grieve, P. G., Isler, J. R., and Sahni, R. (2016). Vital signs and their cross-correlation in sepsis and NEC: a study of 1,065 very-low-birth-weight infants in two NICUs. *Pediatric Research*, 81:315.

Finley, A. O. (2011). Comparing spatially-varying coefficients models for analysis of ecological data with non-stationary and anisotropic residual dependence. *Methods in Ecology and Evolution*, 2(2):143–154.

Finley, A. O., Sang, H., Banerjee, S., and Gelfand, A. E. (2009). Improving the performance of predictive process modeling for large datasets. *Computational statistics & data analysis*, 53(8):2873–2884.

Fohner, A. E., Greene, J. D., Lawson, B. L., Chen, J. H., Kipnis, P., Escobar, G. J., and Liu, V. X. (2019). Assessing clinical heterogeneity in sepsis through treatment patterns and machine learning. *Journal of the American Medical Informatics Association*, 26(12):1466–1477.

Folland, G. B. (2016). *A course in abstract harmonic analysis*, volume 29. CRC press.

Forney, G. D. (1973). The viterbi algorithm. *Proceedings of the IEEE*, 61(3):268–278.

Foucher, Y., Mathieu, E., Saint-Pierre, P., Durand, J.-F., and Daurès, J.-P. (2005). A semi-markov model based on generalized weibull distribution with an illustration for hiv disease. *Biometrical Journal: Journal of Mathematical Methods in Biosciences*, 47(6):825–833.

Futoma, J., Hariharan, S., and Heller, K. (2017a). Learning to Detect Sepsis with a Multitask Gaussian Process RNN Classifier. In *Proceedings of the 34th International Conference on Machine Learning - Volume 70*, ICML'17, pages 1174–1182, Sydney, Australia. JMLR.org. event-place: Sydney, NSW, Australia.

Futoma, J., Hariharan, S., Heller, K. A., Sendak, M., Brajer, N., Clement, M., Bedoya, A., and O'Brien, C. (2017b). An Improved Multi-Output Gaussian Process RNN with Real-Time Validation for Early Sepsis Detection. In *Proceedings of the Machine Learning for Health Care Conference, MLHC 2017, Boston, Massachusetts, USA, 18-19 August 2017*, pages 243–254.

Gal, Y., Chen, Y., and Ghahramani, Z. (2015). Latent gaussian processes for distribution estimation of multivariate categorical data.

Gelfand, A. E. and Ghosh, S. K. (1998). Model choice: a minimum posterior predictive loss approach. *Biometrika*, 85(1):1–11.

Gelfand, A. E., Kim, H.-J., Sirmans, C., and Banerjee, S. (2003). Spatial modeling with spatially varying coefficient processes. *Journal of the American Statistical Association*, 98(462):387–396.

Gelfand, A. E., Schmidt, A. M., Banerjee, S., and Sirmans, C. (2004). Nonstationary multivariate process modeling through spatially varying coregionalization. *Test*, 13(2):263–312.

Ghassemi, M., Pimentel, M. A., Naumann, T., Brennan, T., Clifton, D. A., Szolovits, P., and Feng, M. (2015). A Multivariate Timeseries Modeling Approach to Severity of Illness Assessment and Forecasting in ICU with Sparse,

Heterogeneous Clinical Data. *Proceedings of the ... AAAI Conference on Artificial Intelligence. AAAI Conference on Artificial Intelligence*, 2015:446–453.

Gibbs, M. N. (1998). *Bayesian Gaussian processes for regression and classification*. PhD thesis, University of Cambridge.

Gneiting, T., Kleiber, W., and Schlather, M. (2010). Matérn cross-covariance functions for multivariate random fields. *Journal of the American Statistical Association*, 105(491):1167–1177.

Goovaerts, P. (1997). *Geostatistics for natural resources evaluation*. Oxford University Press.

Goulard, M. and Voltz, M. (1992). Linear coregionalization model: tools for estimation and choice of cross-variogram matrix. *Mathematical Geology*, 24(3):269–286.

Gramacy, R. B. and Lee, H. K. H. (2008). Bayesian treed gaussian process models with an application to computer modeling. *Journal of the American Statistical Association*, 103(483):1119–1130.

Guhaniyogi, R., Finley, A. O., Banerjee, S., and Gelfand, A. E. (2011). Adaptive gaussian predictive process models for large spatial datasets. *Environmetrics*, 22(8):997–1007.

Guhaniyogi, R., Finley, A. O., Banerjee, S., and Kobe, R. K. (2013). Modeling complex spatial dependencies: Low-rank spatially varying cross-covariances with application to soil nutrient data. *Journal of agricultural, biological, and environmental statistics*, 18(3):274–298.

Guttorp, P. and Minin, V. N. (2018). *Stochastic modeling of scientific data*. Chapman and Hall/CRC.

Haining, R. (1993). Statistics for spatial data. *Computers and Geosciences*, 19:615–616.

Heinonen, M., Mannerström, H., Rousu, J., Kaski, S., and Lähdesmäki, H. (2016). Non-stationary gaussian process regression with hamiltonian monte carlo. In *Artificial Intelligence and Statistics*, pages 732–740.

Helterbrand, J. D. and Cressie, N. (1994). Universal cokriging under intrinsic coregionalization. *Mathematical Geology*, 26(2):205–226.

Hensman, J., Fusi, N., and Lawrence, N. D. (2013). Gaussian processes for big data. page 282.

Hensman, J., Rattray, M., and Lawrence, N. D. (2012). Fast variational inference in the conjugate exponential family. pages 2888–2896.

Higdon, D., Swall, J., and Kern, J. (1999). Non-stationary spatial modeling. *Bayesian statistics*, 6(1):761–768.

Hinton, G. (2012). Lecture 6e of his coursera class. *Lecture in his Coursera*.

Hobolth, A. and Jensen, J. L. (2005). Statistical inference in evolutionary models of dna sequences via the em algorithm. *Statistical applications in genetics and molecular biology*, 4(1).

Hobolth, A. and Jensen, J. L. (2011). Summary statistics for endpoint-conditioned continuous-time markov chains. *Journal of applied probability*, 48(4):911–924.

Hochreiter, S. and Schmidhuber, J. (1997). Long short-term memory. *Neural computation*, 9(8):1735–1780.

Hoffman, M. D., Blei, D. M., Wang, C., and Paisley, J. (2013). Stochastic variational inference. *The Journal of Machine Learning Research*, 14(1):1303–1347.

Hripcsak, G., Albers, D. J., and Perotte, A. (2015). Parameterizing time in electronic health record studies. *Journal of the American Medical Informatics Association*, 22(4):794–804.

Hughes, J. P., Guttorp, P., and Charles, S. P. (1999). A non-homogeneous hidden markov model for precipitation occurrence. *Journal of the Royal Statistical Society: Series C (Applied Statistics)*, 48(1):15–30.

Jaderberg, M., Simonyan, K., Zisserman, A., et al. (2015). Spatial transformer networks. pages 2017–2025.

Jang, E., Gu, S., and Poole, B. (2017). Categorical reparametrization with gumble-softmax.

Jenson, A. (1953). Markov chains as an aid in the study of markov processes. *Skand. Aktuarietidskr.*

Jung, K. and Shah, N. H. (2015). Implications of non-stationarity on predictive modeling using EHRs. *Journal of Biomedical Informatics*, 58:168 – 174.

Khan, M., Mohamed, S., Marlin, B., and Murphy, K. (2012). A stick-breaking likelihood for categorical data analysis with latent gaussian models. In Lawrence, N. D. and Girolami, M., editors, *Proceedings of the Fifteenth International Conference on Artificial Intelligence and Statistics*, volume 22 of *Proceedings of Machine Learning Research*, pages 610–618, La Palma, Canary Islands. PMLR.

Khreich, W., Granger, E., Miri, A., and Sabourin, R. (2010). On the memory complexity of the forward-backward algorithm. *Pattern Recognition Letters*, 31(2):91 – 99.

Kingma, D. P. and Ba, J. (2014). Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Kingma, D. P. and Ba, J. (2015). Adam: A method for stochastic optimization.

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

Kitagawa, G. (1987). Non-gaussian state-space modeling of nonstationary time series. *Journal of the American Statistical Association*, 82(400):1032–1041.

Kleiber, W. and Nychka, D. (2012). Nonstationary modeling for multivariate spatial processes. *Journal of Multivariate Analysis*, 112:76–91.

Klompas, M. and Rhee, C. (2016). The cms sepsis mandate: right disease, wrong measure. *Annals of internal medicine*, 165(7):517–518.

Krishnamurthy, V., Leoff, E., and Sass, J. (2018). Filterbased stochastic volatility in continuous-time hidden markov models. *Econometrics and Statistics*, 6:1 – 21. Statistics of Extremes and Applications.

Kuang, Z., Peissig, P., Santos Costa, V., Maclin, R., and Page, D. (2017). Pharmacovigilance via baseline regularization with large-scale longitudinal observational data. pages 1537–1546.

Lasko, T. A. (2014). Efficient Inference of Gaussian-Process-Modulated Renewal Processes with Application to Medical Event Data. *Uncertainty in artificial intelligence : proceedings of the ... conference. Conference on Uncertainty in Artificial Intelligence*, 2014:469–476.

Lasko, T. A. (2015). Nonstationary Gaussian Process Regression for Evaluating Clinical Laboratory Test Sampling Strategies. *Proceedings of the ... AAAI Conference on Artificial Intelligence. AAAI Conference on Artificial Intelligence*, 2015:1777–1783.

Lasko, T. A., Denny, J. C., and Levy, M. A. (2013). Computational Phenotype Discovery Using Unsupervised Feature Learning over Noisy, Sparse, and Irregular Clinical Data. *PLOS ONE*, 8(6):1–13.

Lawrence, N. D. (2004). Gaussian process latent variable models for visualisation of high dimensional data. In *Advances in neural information processing systems*, pages 329–336.

Lawrence, N. D. and Moore, A. J. (2007). Hierarchical gaussian process latent variable models. In *Proceedings of the 24th International Conference on Machine Learning*, ICML '07, pages 481–488, New York, NY, USA. ACM.

Lawrence, N. D. and Quiñonero Candela, J. (2006). 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. ACM.

Lázaro-Gredilla, M., Quiñonero Candela, J., Rasmussen, C. E., and Figueiras-Vidal, A. R. (2010). Sparse spectrum gaussian process regression. *J. Mach. Learn. Res.*, 11:1865–1881.

Li, L., Fu, W., Guo, F., Mowry, T. C., and Faloutsos, C. (2008). Cut-and-stitch: efficient parallel learning of linear dynamical systems on smps. In *Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 471–479. ACM.

Li, S. C.-X. and Marlin, B. (2016). A Scalable End-to-end Gaussian Process Adapter for Irregularly Sampled Time Series Classification. In *Proceedings of the 30th International Conference on Neural Information Processing Systems*, NIPS'16, pages 1812–1820, USA. Curran Associates Inc. event-place: Barcelona, Spain.

Liu, D. C. and Nocedal, J. (1989). On the limited memory bfgs method for large scale optimization. *Math. Program.*, 45(1-3):503–528.

Liu, Y.-Y., Ishikawa, H., Chen, M., Wollstein, G., Schuman, J. S., and Rehg, J. M. (2013). Longitudinal modeling of glaucoma progression using 2-dimensional continuous-time hidden markov model. pages 444–451.

Liu, Y.-Y., Li, S., Li, F., Song, L., and Rehg, J. M. (2015a). Efficient learning of continuous-time hidden markov models for disease progression. In Cortes, C., Lawrence, N. D., Lee, D. D., Sugiyama, M., and Garnett, R., editors, *Advances in Neural Information Processing Systems 28*, pages 3600–3608. Curran Associates, Inc.

Liu, Y.-Y., Li, S., Li, F., Song, L., and Rehg, J. M. (2015b). Efficient learning of continuous-time hidden markov models for disease progression. In *Advances in neural information processing systems*, pages 3600–3608.

Llera, A. and Beckmann, C. F. (2016). Estimating an Inverse Gamma distribution. *ArXiv e-prints*.

Lu, S. (2017). A continuous-time hmm approach to modeling the magnitude-frequency distribution of earthquakes. *Journal of Applied Statistics*, 44(1):71–88.

Luttinen, J. and Ilin, A. (2009). Variational gaussian-process factor analysis for modeling spatio-temporal data. In *Advances in neural information processing systems*, pages 1177–1185.

MacKay, D. J. C. (1992). Bayesian interpolation. *Neural Computation*, 4(3):415–447.

Maclaurin, D. (2016). *Modeling, Inference and Optimization with Compposable Differentiable Procedures*. PhD dissertation, Harvard University.

Maddala, G. S. and Wu, S. (1999). A comparative study of unit root tests with panel data and a new simple test. *Oxford Bulletin of Economics and statistics*, 61(S1):631–652.

Majumdar, A. and Gelfand, A. E. (2007). Multivariate spatial modeling for geostatistical data using convolved covariance functions. *Mathematical Geology*, 39(2):225–245.

Majumdar, A., Paul, D., and Bautista, D. (2010). A generalized convolution model for multivariate nonstationary spatial processes. *Statistica Sinica*, pages 675–695.

Mardia, K. V. and Goodall, C. R. (1993). Spatial-temporal analysis of multivariate environmental monitoring data. *Multivariate environmental statistics*, 6(76):347–385.

Mdzinarishvili, T. and Sherman, S. (2010). Weibull-like model of cancer development in aging. *Cancer informatics*, 9:CIN–S5460.

Meng, R., Saade, S., Kurtek, S., Berger, B., Brien, C., Pillen, K., Tester, M., and Sun, Y. (2017). Growth curve registration for evaluating salinity tolerance in barley. *Plant methods*, 13(1):18.

Merity, S., Shirish Keskar, N., and Socher, R. (2017). Regularizing and Optimizing LSTM Language Models. *arXiv e-prints*, page arXiv:1708.02182.

Metzner, P., Horenko, I., and Schütte, C. (2007). Generator estimation of markov jump processes based on incomplete observations nonequidistant in time. *Physical Review E*, 76(6):066702.

Neal, R. M. (1996). *Bayesian Learning for Neural Networks*. Springer-Verlag, Berlin, Heidelberg.

Nicolau, J. (2014). A new model for multivariate markov chains. *Scandinavian Journal of Statistics*, 41(4):1124–1135.

Paciorek, C. J. and Schervish, M. J. (2004). Nonstationary covariance functions for gaussian process regression. In *Advances in neural information processing systems*, pages 273–280.

Paciorek, C. J. and Schervish, M. J. (2006). Spatial modelling using a new class of nonstationary covariance functions. *Environmetrics: The official journal of the International Environmetrics Society*, 17(5):483–506.

Paroli, R. and Spezia, L. (2008). Bayesian inference in non-homogeneous markov mixtures of periodic autoregressions with state-dependent exogenous variables. *Computational Statistics & Data Analysis*, 52(5):2311–2330.

Pelletier, B., Dutilleul, P., Larocque, G., and Fyles, J. W. (2004). Fitting the linear model of coregionalization by generalized least squares. *Mathematical Geology*, 36(3):323–343.

Qi, Y. and Minka, T. P. (2002). Hessian-based markov chain monte-carlo algorithms. Microsoft.

Quiñonero-Candela, J. and Rasmussen, C. E. (2005). A unifying view of sparse approximate gaussian process regression. *Journal of Machine Learning Research*, 6(Dec):1939–1959.

Rabiner, L. R. (1989). A tutorial on hidden markov models and selected applications in speech recognition. *Proceedings of the IEEE*, 77(2):257–286.

Rasmussen, C. E. and Ghahramani, Z. (2002). Infinite mixtures of gaussian process experts. In *Advances in neural information processing systems*, pages 881–888.

Rasmussen, C. E. and Williams, C. K. I. (2005). *Gaussian Processes for Machine Learning (Adaptive Computation and Machine Learning)*. The MIT Press.

Remes, S., Heinonen, M., and Kaski, S. (2017). Non-stationary spectral kernels. In *Advances in Neural Information Processing Systems*, pages 4642–4651.

Saatçi, Y. (2012). *Scalable inference for structured Gaussian process models*. PhD thesis, University of Cambridge.

Schreiber, T. (1997). Detecting and Analyzing Nonstationarity in a Time Series Using Nonlinear Cross Predictions. *Phys. Rev. Lett.*, 78(5):843–846.

Schulam, P. and Saria, S. (2017a). Reliable decision support using counterfactual models. *Advances in neural information processing systems*, 2017-December:1698–1709.

Schulam, P. and Saria, S. (2017b). What-If Reasoning using Counterfactual Gaussian Processes. In *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, 4-9 December 2017, Long Beach, CA, USA*, pages 1696–1706.

Schulam, P., Wigley, F., and Saria, S. (2015). Clustering longitudinal clinical marker trajectories from electronic health data: Applications to phenotyping and endotype discovery. In *Twenty-Ninth AAAI Conference on Artificial Intelligence*.

Seeger, M. (2003). Pac-bayesian generalisation error bounds for gaussian process classification. *J. Mach. Learn. Res.*, 3:233–269.

Seymour, C. W., Liu, V. X., Iwashyna, T. J., Brunkhorst, F. M., Rea, T. D., Scherag, A., Rubenfeld, G., Kahn, J. M., Shankar-Hari, M., Singer, M., et al. (2016). Assessment of clinical criteria for sepsis: for the third international consensus definitions for sepsis and septic shock (sepsis-3). *Jama*, 315(8):762–774.

Shen, X., Su, H., Niu, S., and Demberg, V. (2018). Improving variational encoder-decoders in dialogue generation.

Simpson, S. E., Madigan, D., Zorych, I., Schuemie, M. J., Ryan, P. B., and Suchard, M. A. (2013). Multiple self-controlled case series for large-scale longitudinal observational databases. *Biometrics*, 69(4):893–902.

Smyth Gordon, K. (2004). Linear models and empirical bayes methods for assessing differential expression in microarray experiments. *Statistical Applications in Genetics and Molecular Biology*, 3(1):1–25.

Snelson, E. and Ghahramani, Z. (2006a). Sparse gaussian processes using pseudo-inputs. In *Advances in neural information processing systems*, pages 1257–1264.

Snelson, E. and Ghahramani, Z. (2006b). Sparse gaussian processes using pseudo-inputs. In Weiss, Y., Schölkopf, B., and Platt, J. C., editors, *Advances in Neural Information Processing Systems 18*, pages 1257–1264. MIT Press.

Snelson, E. and Ghahramani, Z. (2007). Local and global sparse gaussian process approximations. In *Artificial Intelligence and Statistics*, pages 524–531.

Snelson, E., Ghahramani, Z., and Rasmussen, C. E. (2004a). Warped gaussian processes. In *Advances in neural information processing systems*, pages 337–344.

Snelson, E., Ghahramani, Z., and Rasmussen, C. E. (2004b). Warped Gaussian Processes. In Thrun, S., Saul, L. K., and Schölkopf, B., editors, *Advances in Neural Information Processing Systems 16*. MIT Press.

Sonnenberg, F. A. and Robert, B. J. (1993). Markov models in medical decision making: A practical guide. *Med Decis Making*, 13:322–338.

Stein, M. L. (2005). Nonstationary spatial covariance functions. *Unpublished technical report*.

Subakan, C., Traa, J., and Smaragdis, P. (2014). Spectral learning of mixture of hidden markov models. In Ghahramani, Z., Welling, M., Cortes, C., Lawrence, N. D., and Weinberger, K. Q., editors, *Advances in Neural Information Processing Systems 27*, pages 2249–2257. Curran Associates, Inc.

Sun, Y. and Genton, M. G. (2011). Functional boxplots. *Journal of Computational and Graphical Statistics*, 20(2):316–334.

Tank, A., Fox, E. B., and Shojaie, A. (2017). Granger Causality Networks for Categorical Time Series. *ArXiv e-prints*.

Tataru, P. and Hobolth, A. (2011). Comparison of methods for calculating conditional expectations of sufficient statistics for continuous time markov chains. *BMC bioinformatics*, 12(1):465.

Thwaites, P. (2013). Causal identifiability via Chain Event Graphs. *Artificial Intelligence*, 195:291 – 315.

Tipping, M. E. and Bishop, C. (1999). Probabilistic principal component analysis. *Journal of the Royal Statistical Society, Series B*, 21(3):611–622. Available from http://www.ncrg.aston.ac.uk/Papers/index.html.

Titman, A. C. and Sharples, L. D. (2008). A general goodness-of-fit test for Markov and hidden Markov models. *Statistics in Medicine*, 27(12):2177–2195.

Titsias, M. (2009). Variational learning of inducing variables in sparse gaussian processes. In van Dyk, D. and Welling, M., editors, *Proceedings of the Twelth International Conference on Artificial Intelligence and Statistics*, volume 5 of *Proceedings of Machine Learning Research*, pages 567–574, Hilton Clearwater Beach Resort, Clearwater Beach, Florida USA. PMLR.

Titsias, M. and Lawrence, N. D. (2010). Bayesian gaussian process latent variable model. In Teh, Y. W. and Titterington, M., 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. PMLR.

Tolvanen, V., Jylänki, P., and Vehtari, A. (2014). Expectation propagation for nonstationary heteroscedastic gaussian process regression. In *2014 IEEE International Workshop on Machine Learning for Signal Processing (MLSP)*, pages 1–6. IEEE.

Tresp, V. (2000). A bayesian committee machine. *Neural Computation*, 12(11):2719–2741.

Tresp, V. (2001). Mixtures of gaussian processes. In *Advances in neural information processing systems*, pages 654–660.

Ursin, G., Sen, S., Mottu, J.-M., and Nygård, M. (2017). Protecting privacy in large datasets—first we assess the risk; then we fuzzy the data. *Cancer Epidemiology and Prevention Biomarkers*, 26(8):1219–1224.

Urtasun, R. and Darrell, T. (2007). Discriminative gaussian process latent variable model for classification. In *Proceedings of the 24th international conference on Machine learning*, pages 927–934.

Van Loan, C. (1978). Computing integrals involving the matrix exponential. *IEEE transactions on automatic control*, 23(3):395–404.

Ver Hoef, J. M. and Barry, R. P. (1998). Constructing and fitting models for cokriging and multivariable spatial prediction. *Journal of Statistical Planning and Inference*, 69(2):275–294.

Ver Hoef, J. M., Cressie, N., and Barry, R. P. (2004). Flexible spatial models for kriging and cokriging using moving averages and the fast fourier transform (fft). *Journal of Computational and Graphical Statistics*, 13(2):265–282.

Wackernagel, H. (2013). *Multivariate geostatistics: an introduction with applications*. Springer Science & Business Media.

Wang, C. and Blei, D. M. (2013). Variational inference in nonconjugate models. *Journal of Machine Learning Research*, 14(Apr):1005–1031.

Wang, J., Hertzmann, A., and Fleet, D. J. (2006). Gaussian process dynamical models. In Weiss, Y., Schölkopf, B., and Platt, J. C., editors, *Advances in Neural Information Processing Systems 18*, pages 1441–1448. MIT Press.

Wang, J. M., Fleet, D. J., and Hertzmann, A. (2008). Gaussian process dynamical models for human motion. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 30(2):283–298.

Wang, Q. (2012). Kernel Principal Component Analysis and its Applications in Face Recognition and Active Shape Models. *ArXiv e-prints*.

Wang, X., Sontag, D., and Wang, F. (2014). Unsupervised learning of disease progression models. In *Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '14, pages 85–94, New York, NY, USA. ACM.

Wei, W., Wang, B., and Towsley, D. (2002). Continuous-time hidden markov models for network performance evaluation. *Performance Evaluation*, 49(1):129 – 146. Performance 2002.

Wheeler, D. C. and Calder, C. A. (2007). An assessment of coefficient accuracy in linear regression models with spatially varying coefficients. *Journal of Geographical Systems*, 9(2):145–166.

Wilson, A. G. (2014). *Covariance kernels for fast automatic pattern discovery and extrapolation with Gaussian processes*. PhD thesis, University of Cambridge.

Yen, A. M., Chen, T. H., Duffy, S. W., and Chen, C.-D. (2010). Incorporating frailty in a multi-state model: application to disease natural history modelling of adenoma-carcinoma in the large bowel. *Statistical Methods in Medical Research*, 19(5):529–546. PMID: 20488838.

Zeifman, A. and Isaacson, D. L. (1994). On strong ergodicity for nonhomogeneous continuous-time markov chains. *Stochastic Processes and their Applications*, 50(2):263 – 273.

Zeiler, M. D. (2012). ADADELTA: An Adaptive Learning Rate Method. *ArXiv e-prints*.

Zhang, K., Chao, W.-L., Sha, F., and Grauman, K. (2016). Video summarization with long short-term memory. pages 766–782.