Gaussian Process Based Approaches for Survival Analysis

Alan D. Saul

Department of Computer Science
University of Sheffield

A thesis submitted in partial fulfilment of the requirements for the degree of
Doctor of Philosophy

August 2017

I dedicate this thesis to my loving family.

Acknowledgements

There are many people who must be thanked for their support throughout the years.

First of all I must thank my supervisor, Prof. Neil Lawrence, whose advice and support has been invaluable. He has provided an environment in which it has been a pleasure to grow, and a feeling of family within an excellent research group.

A number of other people have provided me with opportunities to learn throughout the years and for whose help I am immensely thankful, in particular two people stand out. James Hensman, who taught me so much of what I now know with stimulating conversations and discussions of exciting new ideas throughout my studies, aswell as always provided a friendship and helping hand. Aki Vehtari, must also be personally thanked, he has guided me through the area of survival analysis within the domain of Gaussian processes; whilst also providing many entertaining evenings enjoying fantastic beer and food all around the world, he has always made me feel very welcome.

I would like to thank my friends and colleagues from the Sheffield ML research group as well as the numerous visitors, all of which have made it a pleasure to study in Sheffield, whos friendships have made it a pleasure to study in Sheffield and who have taught me a great number of things. As a constantly changing group there are too many people to name, but to name a few; Alessandra Tosi, Andreas Damianou, Arifur Rahman, Fariba Yousefi, Javier González, Luisa Cutillo, Mauricio A. Álvarez, Max Zwießele, Mike Croucher, Mike Smith, Nicolas Durrande, Nicolo Fusi, Teo De Campos, Zhenwen Dai, and visitors Alex Grigorievskiy, Arno Solin, Carl Henrik Ek, Cristian Guarnizo, César Lincoln, Francisco Ruiz.

I would like to thank the NIPS and AISTATS community of anonymous reviewers, Amazon for a generous donation of AWS compute time, and the University of Sheffield Faculty of Engineering for providing my studentship.

Finally I would like to thank my girlfriend Bodil Oudshoorn, who has supported me through the final years of my study.

Abstract

Traditional machine learning focuses on the situation where a fixed number of features are available for each data-point. For medical applications each individual patient will typically have a different set of clinical tests associated with them. This results in a varying number of observed per patient features. An important indicator of interest in medical domains is survival information. Survival data presents its own particular challenges such as censoring. The aim of this thesis is to explore how machine learning ideas can be transferred to the domain of clinical data analysis. We consider two primary challenges; firstly how survival models can be made more flexible through non-linearisation and secondly methods for missing data imputation in order to handle the varying number of observed per patient features. We use the framework of Gaussian process modelling to facilitate conflation of our approaches; allowing the dual challenges of survival data and missing data to be addressed. The results show promise, although challenges remain. In particular when a large proportion of data is missing, greater uncertainty in inferences results. Principled handling of this uncertainty requires propagation through any Gaussian process model used for subsequent regression.

Table of contents

List of figures

List of tables

Nomenclature

Approximations

q	Approximate distribution
$\boldsymbol{\Sigma}_V$	Variational covariance parameters
$\boldsymbol{\Omega}_V$	Diagonal variational covariance parameters
$\boldsymbol{A}$	Laplace approximation precision matrix
$\boldsymbol{W}$	Negative Hessian of likelihood
$\boldsymbol{\mu}_{fu}$	Mean of approximate distribution $q(\mathbf{u}_f)$
$\boldsymbol{\mu}_{gu}$	Mean of approximate distribution $q(\mathbf{u}_g)$
$\boldsymbol{\mu}_V$	Variational mean parameters
$\boldsymbol{S}_f$	Covariance of approximate distribution $q(\mathbf{u}_f)$
$\boldsymbol{S}_g$	Covariance of approximate distribution $q(\mathbf{u}_g)$
$\boldsymbol{\theta}_V$	Vector of variational parameters

Missing

$\mathbf{M}$	Matrix of indices that indicate missing values
n_j	Number of observations in dimension j of $\mathbf{Y}$ or $\boldsymbol{S}$
$\mathcal{O}_j$	Considering only indices of non-missing values in dimension j of $\mathbf{Y}$ or $\boldsymbol{S}$
$\mathbf{Y}^{\mathcal{M}}$	Set of missing observations
$\mathbf{Y}^{\mathcal{O}}$	Set of non-missing observations

Chained GP

k_g	Kernel function for latent function g
$\mathbf{K}_{\boldsymbol{gg}}$	Kernel function for g, evaluated $k(\mathbf{X}, \mathbf{X})$
$\mathbf{K}_{\boldsymbol{g}\mathbf{u}_g}$	Kernel function for g, evaluated $k(\mathbf{X}, \mathbf{Z})$
$\mathbf{Q}_{\boldsymbol{gg}}$	Nystrom approximation to covariance matrix $\mathbf{K}_{\boldsymbol{gg}}$
$\mathbf{u}_f$	Inducing points corresponding to function f
$\mathbf{u}_g$	Inducing points corresponding to function g

Functions

w	Coefficient for regression
f	Generic latent mapping function 1
g	Generic latent mapping function 2
λ^{-1}	Inverse link function / transformation function
υ	Generic latent mapping function 3
λ	Link function
μ	Mean

Gaussian process

$\boldsymbol{D}$	BGPLVM precision matrix
$\mathrm{GP}\left(\mu, \mathbf{K}\right)$	Sample from Gaussian process
k_f	Kernel function for latent function f
$\mathbf{K}_{\boldsymbol{ff}}$	Kernel function for f, evaluated $k(\mathbf{X}, \mathbf{X})$
$\mathbf{K}_{\boldsymbol{f^*f}}$	Kernel function for f, evaluated predictive points $\mathbf{X}^*$, $k(\mathbf{X}^*, \mathbf{X})$
$\mathbf{K}_{\boldsymbol{f}\mathbf{u}}$	Kernel function for f, evaluated $k(\mathbf{X}, \mathbf{Z})$
$\hat{\mathbf{K}}$	Reformulated BGPLVM covariance matrix
$\boldsymbol{\theta}_K$	Hyperparameters of kernel

$\boldsymbol{\psi}_1$	$\mathbb{E}_{q(\mathbf{X}\vert\boldsymbol{\theta}_V)}\Big[\mathbf{K}_{\boldsymbol{f}\mathbf{u}}\Big]$
$\boldsymbol{\psi}_2$	$\mathbb{E}_{q(\mathbf{X}\vert\boldsymbol{\theta}_V)}\Big[\sum_{i=1}^{n}\mathbf{K}\mathbf{u}, \boldsymbol{f}_i\mathbf{K}\boldsymbol{f}_i, \mathbf{u}\Big]$
ψ_0	$\mathbb{E}_{q(\mathbf{X}\vert\boldsymbol{\theta}_V)}\Big[\mathrm{tr}\left(\mathbf{K}_{\boldsymbol{f}\boldsymbol{f}}\right)\Big]$
$\mathbf{Q}_{\mathbf{ff}}$	Nystrom approximation to covariance matrix $\mathbf{K}_{\boldsymbol{f}\boldsymbol{f}}$
$\boldsymbol{S}$	Observed output data
$\mathbf{U}$	Inducing points
$\mathbf{Z}$	Inducing inputs

Misc

const	Constant term with respect to a random variable
$\mathbb{E}_{p(A)}\Big[A\Big]$	Expectation of random variable A, under distribution $p(A)$
$\mathbb{Z}$	Integer space
$\mathbb{Z}_{\geq 0}$	Non-negative integer space
$\mathcal{O}$	Order
$\mathbb{R}$	Real space
$\mathbb{R}_{\geq 0}$	Non-negative real space
$\mathbb{R}_{>0}$	Positive real space
$\mathbb{Z}_{\{0,1\}}$	Set of zero and one

Probability distributions

$\boldsymbol{\Sigma}$	Covariance matrix
L	Likelihood
β^{-1}	Gaussian likelihood variance
ϵ	Noise model
Z	Normalizer term

$\boldsymbol{\theta}_L$	Parameters of likelihood distribution
$\mathbf{\Lambda}$	Precision matrix
π	Probability of Bernoull distribution, or π
α	Scale parameter
β	Shape parameter
Φ	Squashing function

Survival

$h_{0T}(t\|\boldsymbol{\theta})$	Baseline hazard function
$S_{0T}(t\|\boldsymbol{\theta})$	Baseline survival function
ν	Censoring indicator, 0 for censored, 1 for uncensored
M	Censored set of individuals
$F_T(t\|\boldsymbol{\theta})$	Cumulative failure distribution function
τ	Constant latent baseline hazard function within time segment
$\Lambda_{0T}(t\|\boldsymbol{\theta})$	Cumulative baseline hazard function
$\Lambda_T(t\|\boldsymbol{\theta})$	Cumulative hazard function
$f_T(t\|\boldsymbol{\theta})$	Failure density function, probability of failure at any time
$h_T(t\|\boldsymbol{\theta})$	Hazard function, conditional rate of failing at a certain time, conditional on surviving up until that instance
η	Latent hazard function
B	Number of time segments
$h_R(\boldsymbol{\theta}, \mathbf{x})$	Relative hazard function
R	Set of individuals at risk of failure
T	Random variable failure time
$S_T(t\|\boldsymbol{\theta})$	Probability of surviving beyond a certain time

η	Time of termination of clinical trial
δ	Small change in time
s	Time segment
s	True (uncensored) failure time
t	Instantiation of failure time
$\mathbf{t}$	Vector of failure times
K	Uncensored set of individuals
$\mathbf{X}$	Input data or covariates
$\mathbf{Y}$	Observed or latent output data

AFT	Accelerated failure time
ARD	Automatic relevance determination
BGPLVM	Bayesian Gaussian process latent variable model
CCD	Central composite design
CPH	Cox Proportional Hazards
DPPCA	Dual probablistic principle component analysis
DTC	Deterministic training conditional
ELBO	Evidence lower bound
EP	Expectation Propagation
FITC	Fully independent training conditional
GLM	Generalised additive model
GLM	Generalised linear model
GP	Gaussian process
GPLVM	Gaussian process latent variable model

HR	Hazard ratio
i.i.d	Independent and identically distributed
KL	Kullback-Leibler
KM	Kaplan Meier
LU	Lower Upper
MAE	Mean absolute error
MAP	Maximum a posteriori
MAE	Missing at random
MBI	Model based imputation
MCAR	Missing completely at random
MC	Monte Carlo
MCMC	Markov Chain Monte Carlo
MICE	Multiple imputation by chained equations
MI	Multiple imputation
MNAR	Missing not at random
NLPD	Negative log predictive density
PCA	Principle component analysis
PMM	Predictive mean matching
PPCA	Probablistic principle component analysis
RBF	Radial basis function
SI	Single imputation

Symbol And Matrix Notation

Lowercase symbols, x, will refer to scalar values. Bold lowercase symbols, $\mathbf{x}$, will denote used for vectors of scalar values. Bold uppercase symbols will denote matrices with multiple dimensions. The notation $\mathbf{x}_{i,:}, \mathbf{x}_{:,j}, x_{i,j}$ will refer to the ith row, the jth column, and i, jth element respectively of a matrix $\mathbf{X}$, and similarly so for any other matrix used in this work.

1

Introduction

Proper handling of uncertainty is a cornerstone of modern data analysis. Survival analysis is the study of time-to-event data, or classically *failure time* data. Within medical statistics the consequences of making incorrect predictions may cause unnecessary stress or even harm. Survival analysis is also applied within the area of engineering, particularly when assessing the reliability of equipment under stress. Having an accurate measure of the uncertainty associated with predictions of how long an individual or equipment, may survive is critical for a number of reasons. For example, interventions based on predictions may be expensive or dangerous and should not be undertaken if one is largely unsure about the true outcome. Typically in survival analysis regression, the prediction of failure time is believed to be an unknown function of some observed, or partially observed, covariates. Analysis of such data is often complicated by *censoring* of the time at which a failure occurs; this may be as a result of a patient leaving a clinical trial, or the clinical trial being forced to terminate before the failure is observed. The Bayesian probabilistic framework of Gaussian processes provides a principled way in which to handle the uncertainty associated with unknown functions. A distribution over possible functions is inferred by combining prior knowledge about the function, with the information provided by a set of observations. In its standard form, Gaussian processes make a normality assumption that means they cannot be trivially applied to survival analysis regression. Luckily in recent years a huge array of approximate inference methods have provided a number of ways in which this problem may be sidestepped.

Gaussian processes have been popular over the last few decades within the area of geostatistics where the model is known as *kriging*. In recent years the framework has become a popular choice for a range of other machine learning problems. Machine learning models are typically models that can learn some characteristic of interest without relying on a fixed set of rules, instead learning directly from provided data, though its definition is loose. As a result of an increased interest in machine learning models, their application to survival analysis regression has naturally been considered. Some of the most popular machine learning methods that have been used for survival analysis regression include neural networks (Rumelhart et al., 1985), decision trees, random forests (Breiman, 2001), and kernel methods such as kernel ridge regression (Saunders et al., 1998) and support vector regression (Vapnik, 1998).

1.1 Machine Learning Approaches To Survival Analysis

It will prove useful to consider the advantages and disadvantages of the Gaussian process framework that is the focus of this thesis, relative to the existing machine learning methods that have previously been used for survival analysis regression. Beyond of the context of this thesis, machine learning models have also been used for variable selection for survival analysis (Fan et al., 2005). This problem considers picking a few of the model relevant clinical variables relating to a particular outcome, when a large number of variables are present, e.g. genes. The models proposed at the end of this thesis may also be a useful approach for tackling these problems, though this hypothesis has not been tested.

1.1.1 Neural Networks

Neural network methods, and similarly deep neural network methods have seen a resurgence in popularity in the last decade. This has primarily been as a result of increased compute power, increased dataset sizes, as well as a number of algorithmic innovations that have eased issues with initialisation and optimisation (Goodfellow et al., 2016; Kingma and Ba, 2014). One of the advantages of neural network methods is that they allow arbitrarily complicated functions to be represented, and consequently they can be used to fit non-linear functions that may arise in survival analysis, as will be discussed in Chapter 2. This flexibility however often proves to be a mixed blessing. As a result, the models can be susceptible to overfitting if care is not taken, though regularisation techniques are now being used to address this (Srivastava et al., 2014). The large number of variables (weights of the network) that need to adjusted also means that there is likely to be an array of local minima, and saddle points that need to be circumvented if the optimal model is to be found (Dauphin et al., 2014).

Robust optimisation of these models has been, and continues to be, one of the major obstacles to applying neural network models. Neural networks however have been successfully used for survival analysis in the literature (Biganzoli et al., 1998; Ripley and Ripley, 2001). Unfortunately, traditional neural networks typically do not naturally consider the uncertainty in the variables being learnt (weights) and as a result do not provide a faithful representation of the uncertainty in their predictions. Bayesian neural networks (Neal, 1996), where prior uncertainty is specified for the weights values, go some way to alleviating both the optimisation issues and predictive uncertainty, and these models have been considered in the context of survival analysis regression (Bakker et al., 2000). Though the uncertainty can then be considered, inference of this posterior uncertainty still must be undertaken. Traditionally a popular approach has been Markov Chain Monte Carlo (MCMC) which can often have difficulties when datasets become large, as has become common in recent years. Exciting new innovations in variational inference (Ranganath et al., 2014) are also being applied to Bayesian neural networks to rectify this shortcoming, though its applications to survival analysis are still in its early stages, with one notable exception (Ranganath et al., 2016).

The major concern when using neural network models, particularly deep neural networks, for survival analysis is that in general they are very difficult to interpret. Though the method of learning is well understood, the resulting model is typically treated as a black-box, and the underlying rules that have been learnt are not clear. This can pose a serious issue when these models are to be used in a clinical setting, including survival analysis. Interpreting how a model comes to a specific conclusion is often important both as a safety measure, and also as the effect of individual variables is often of interest in itself. In contrast, with Gaussian process models it is typically relatively simple to visualise the corresponding effect of each input variable on the output prediction, and it is also possible to introduce prior knowledge about the form of the function through the specification of the kernel (Section 3.1.1).

1.1.2 Decision Trees And Random Forests

Decision trees in contrast to neural networks are simple to interpret, as ultimately they learn a series of simple rules in a tree-like graph that lead to a prediction. These rules are easily accessible due to the specification of the model. One limitation of decision trees is that predictions are typically made based on rules with a hard binary decisions, and so in practice regression problems tend to show degraded performance where we would normally expect the underlying function to be smooth (Hastie et al., 2009). One of decision trees other major shortcomings is that they are known to often generate overly complex models, and hence don't tend to generalise well from training data. This shortcoming also manifests itself by not being very robust — a small change in the training data can result in a major change in the resulting model (Hastie et al., 2009). For survival analysis regression, where serious interventions may occur as a result of the predictions of the model, this lack of robustness is clearly a undesirable feature; a property that Gaussian processes do not typically share. Random forests attempt to alleviate this robustness issue by learning many different decision trees, and averaging their results, though this does come at the expense of some interpretability. Despite these limitations, decision trees, random forests and their Bayesian counterparts have been used for survival analysis with relative success (Ishwaran et al., 2008; Sefrioui et al., 2017).

1.1.3 Kernel Ridge Regression

Generalised linear regression has been the dominant method of choice for survival analysis regression, though it does require a number of modelling assumptions that are often not true in practice, as is discussed in Chapter 2. One of the most prominent shortcomings is its widely used linear assumptions about the effect of the covariates. Ridge regression (Hoerl and Kennard, 1970) assigns a prior over the weights used for the linear assumption and finds a *maximum a posteriori* (MAP) solution (Murphy, 2012). Kernel ridge regression (Saunders et al., 1998) attempts to rectify the linear assumption through the use of kernel functions to project the input into a different feature space, this allows non-linear functions to be learnt. This relaxation of the linear function is one major benefit of these models when they are used for survival analysis regression (Cawley et al., 2006b). Kernel ridge regression, similarly to linear regression, only consider the mean of the predicted function; often it is crucial to also consider the variance around this prediction when making use of predictions. From the weight-space view, a Gaussian process can be seen as the Bayesian counterpart to kernel ridge regression, where not only a MAP solution is sought for the weights, but also the posterior uncertainty (Rasmussen and Williams, 2006). By additionally considering the uncertainty around the weights, predictive uncertainty capabilities are also naturally provided, though some extensions to kernel ridge regression attempt to also rectify this shortcoming (Cawley et al., 2006a). As previously stated, faithfully representing uncertainty in the predictions in some cases may be important, particularly when expensive actions may be taken in response to the prediction. The sparseness of the solutions given by some kernel ridge regression methods can however result in a lower computational burden when compared with standard Gaussian process regression.

1.1.4 Support Vector Regression

The class of support vector regression (SVR) models (Vapnik, 1998) provide additional kernel methods that can be used for non-linear functions within a survival analysis regression setting (Khan and Zubek, 2008; Shivaswamy et al., 2007; Van Belle et al., 2011). The models, similarly to kernel ridge regression and Gaussian processes, utilise kernels to allow non-linear functions to be represented. In contrast to kernel ridge regression and Gaussian processes, SVR models modify the underlying model in a way that is unnatural from a probabilistic standpoint. This modification is made in order to use quadratic program solvers that can reduce the computational workload required for inference; computational savings over Gaussian process models have also been shown to occur in some survival analysis settings (Shivaswamy et al., 2007). During prediction in particular, these methods can be much faster than Gaussian process methods. SVR models additionally do not provide a realistic probabilistic output, though heuristic approaches are often used to produce a probabilistic output, these heuristics cannot be well justified from a Bayesian standpoint (Murphy, 2012), similarly to the heuristic approaches used in vanilla neural network methods. Throughout this work we wish to maintain a strong focus on the uncertainty associated with the survival analysis predictions, we will not consider support vector methods in more detail.

1.2 Outline Of This Thesis

This thesis attempts to unify the classical survival analysis branch of statistics with the modern modelling advances in the Gaussian process area of machine learning. The previous section attempts to justify the focus on this particular machine learning framework. Chapter 2 introduces the basic concepts of survival analysis, showing that many of the most popular methods share a restrictive generalised linear modelling assumption with respect to the covariates used for prediction of survival. Chapter 3 provides a review of standard uses of Gaussian processes, including existing approximations for handling non-normally distributed data, and scalable Gaussian process inference. Chapter 4 briefly discusses existing models that attempt to unify these two areas, and then provides the first main contribution, based on Saul et al. (2016); a more flexible non-linear Gaussian process based survival analysis model that is able to scale, known as the *chained survival analysis model*. The same inference method will be shown to be flexible enough to account for a range of different models. Chapter 5 will consider the importance of missing data imputation, with which clinical data is often plagued. The second major contribution will be presented, a latent variable model for imputation that is able to handle both Gaussian and non-Gaussian output types, such as binary observations. Finally Chapter 6 will discuss the importance of propagating uncertainty associated with the inputs for survival analysis; and provide a method by which this propagation may be performed. The uncertainty associated with the input may arise from the process of imputation provided in the preceding chapter. It will also propose a number of alternative models that could build on the research presented throughout.

2

Survival Analysis

This chapter will introduce the basic principles of survival analysis and assumptions that are frequently made in some of the most popular models in the literature. Commonly used components that form the basis of the field are first derived. A particular focus will be on functions that will be utilised in later chapters, and as such will be convenient to introduce early, namely; failure time, failure density function, cumulative failure; survival function, hazard function and censoring.

The generalised linear model (GLM) interpretation of some of the most popular models in survival analysis regression will be provided. This will make the limitations of these methods transparent, and serve as a springboard for the novel contributions of Chapter 4, in which a number of these limitations are relaxed.

2.1 Failure Time

A *failure time*, or *time-to-event*, is a point event that occurs at least once for each *individual* (subject of analysis). An individual may be a patient, where the failure is death or recurrence of a disease. The individual could alternatively be a machine component that is undergoing testing by inspectors for example, where the event is the onset of some mechanical failure. Throughout this body of work survival analysis is primarily discussed in the context of medical data, and so individuals will usually be assumed to be humans unless otherwise stated. The failure time will be discussed in terms of the death of the individual, though it should be noted that the techniques outlined and developed are applicable to a range of applications, including equipment failure and customer retention analysis (Barlow and Proschan, 1975; Rosset et al., 2003). Measurements of failure time are taken with respect to a *time origin*, for example this may be birth, or it may be entrance into the study, amongst other possibilities. It is the point at which time, $t \in \mathbb{R}_{\geq 0}$, is 0 for the individual, and the failure time will necessarily be after this time. There are differing opinions on the most appropriate time origin. It is common to use the start time of the trial, though it has been shown that in certain cases such as when the time of first incidence (for example onset of disease) is difficult to define, the age of the patient or calender time can lead to more accurate inferences (Sperrin and Buchan, 2012).

Survival analysis approaches the problem of making inferences about failure times. Failure times in this work are assumed to necessarily occur just once, though the time at which it occurs may be known or partially known.

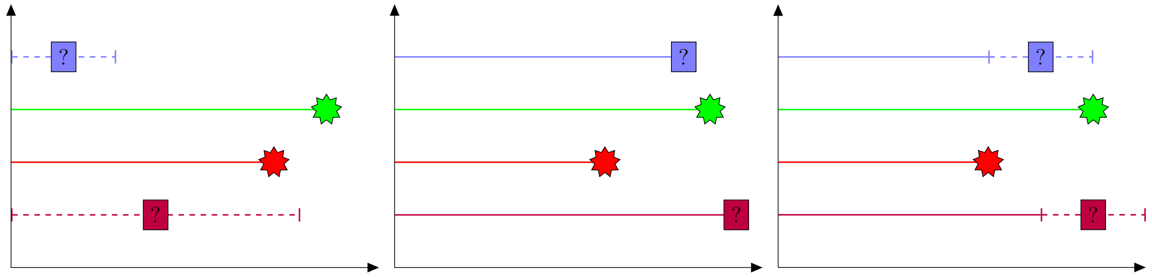
Failure times, if known, should be precisely defined for each individual (Cox, David and Oakes, 1984). Often explanatory variables (otherwise known as covariates) will also be available that differ between individuals, and the task will be then either to infer the implications of such variables, or make predictions using them.

2.1.1 Censored Data

Often in analysis of failure time data, although it is known the failure will occur, it is not known the exact time at which the failure will happen, or has happened. Data in which only partial knowledge of the failure time is available are known as *censored data*. Censored data make the task of modelling the process that created the data significantly more difficult, but it is important to include these data, they cannot be simply discarded, otherwise a bias can be introduced to any inferences made. For example, at the end of a clinical trial (end of follow-up), if the failure hasn't occurred for a subject, the exact time at which the failure happens in the future (beyond the follow-up period) will not be known. Clearly knowledge that the failure hasn't occurred within the follow-up period for this particular individual indicates that some attribute of this the individual is potentially assisting their prolonged survival. It is the task of survival analysis techniques to take this partial knowledge about censored failure times into consideration, as well as the known failure times.

Partial knowledge of the failure time can manifest itself in several ways:

1. *Left censoring* — Knowing the failure occurred before a certain time, but not precisely when; this is not frequently seen in clinical data. One example where it is observed is when checkups are only done at discrete times, and the event occurs between the beginning of the trial and the first checkup.



(a) Left censoring (b) Right censoring (c) Interval censoring

Fig. 2.1 Three types of censoring, left, right and interval. Stars represent an *observed* failure time. Intervals containing a question mark represent a period in which the failure is known to have occurred, but the exact time is *unknown* (censoring has occurred). In the case of right censoring, the event is known to occur beyond the question mark time. Colours represent different patients.

2. *Right censoring* — Knowing the failure occurred after a certain time, but not precisely when; this is very common in clinical data, either at the end of follow-up period, or a patient dropping out of the trial.

3. *Interval censoring* — Knowing the failure occurred within a period of time, but not precise knowledge of the time; this is also very common in clinical data as often it is known the failure occurred between two visits.

Usually the data provided will contain a set of failure times for each of n individuals, $\mathbf{t} \in \mathbb{R}_{\geq 0}^{n\times 1}$, and a set of associated failure indicator variables, $\boldsymbol{\nu} \in \mathbb{Z}_{\{0,1\}}^{n\times 1}$, indicating whether the failure occurred at this time, or whether it was a censoring time, for each individual. For example, in right censoring the failure indicator, ν_i, would indicate whether the failure occurred after t_i or whether the failure occurred at t_i for individual i. Throughout this body of work the value of ν_i will be 1 if the failure occurred at time, t_i, or 0 if t_i was a censoring time.

2.1.2 Failure Density Function

The *failure density function*,

$$f_T(t|\boldsymbol{\theta}) = \lim_{\delta t\to 0} \frac{\Pr(t < T \leq t + \delta t)}{\delta t},$$

is the probability density function of the random variable, failure time, T. Where $\boldsymbol{\theta}$ are parameters required for the chosen density function.

The failure density function provides the probability that the failure time will occur at time t, given no further information.

The exact form of the failure density function is often unknown and so is a target of inference. Many traditional survival analysis approaches specify a parameterised distribution for the failure density function and infer the parameter values that allow the distribution to match the data, these are known as *parametric models* (Collett, 2015). It is most commonly a continuous distribution, that only assigns probability density to the positive domain. The distribution chosen has implications on the flexibility of the model, and an incorrect choice can give rise to poor inference abilities. Common choices are the exponential, gamma, Weibull, log-normal, log-logistic and proportional-hazard family distributions, see Cox, David and Oakes (1984) and Collett (2015) for details on the implications of choosing specific distributions.

The *cumulative failure function*,

$$F_T(t|\boldsymbol{\theta}) = \Pr(T \leq t) = \int_{-\infty}^{t} f(k|\boldsymbol{\theta})dk,$$

is the cumulative probability of the failure time occurring up to and including failure time, t, from the time origin. Since the probability of a failure is always zero or positive, the cumulative failure function is a monotonic function, various survival analysis techniques choose to model this function because of this attribute, such as modelling it as a gamma process (Ibrahim et al., 2005).

2.1.3 Survival Function

It will be often desirable to find the probability that a random failure, T, occurs after a given time, t, this known as the *survival function* which is denoted $S_T(t|\boldsymbol{\theta}) = P(T > t)$. The survival function from the existing definitions can be derived in the following way,

$$\begin{aligned}
S_T(t|\boldsymbol{\theta}) &= 1 - F_T(t|\boldsymbol{\theta}) && \text{(2.1a)}\\
&= 1 - \int_{-\infty}^{t} f(k|\boldsymbol{\theta})dk && \text{(2.1b)}\\
&= \int_{t}^{\infty} f(k|\boldsymbol{\theta})dk\\
&= \Pr(T > t).
\end{aligned}$$

The implications of obtaining a good approximation of the survival function should be obvious; given an accurate survival function it would then be possible to query the probability that a patient will survive beyond any given time.

It will also be convenient for future derivations to format the survival function in terms of the failure density function,

$$\frac{dS_T(t|\boldsymbol{\theta})}{dt} = S'_T(t|\boldsymbol{\theta}) = -f_T(t|\boldsymbol{\theta}). \tag{2.2}$$

2.1.4 Hazard Function

Though the failure density function and the corresponding survival function may be useful, in some cases it may be more interesting to discover the underlying instantaneous failure rate (Wan. Tang, 2012), this is known as the *hazard function*. The cumulative survival and cumulative failure function are monotonic; this is not a requirement of the hazard function. Inference of the hazard function can in some cases be simpler. The hazard function describes the instantaneous *rate* of failure (not probability as the hazard may be greater than one), conditioned on the subject having survived up until time t. The rate is analogical to the probability that the failure will happen in the infinitesimally small time after t, $[t, t+\delta t)$. So $h_T(t|\boldsymbol{\theta})\delta t$ is the approximate probability of failure occurring (Ibrahim et al., 2005) in $[t, t+\delta t)$, given survival up to time t. The hazard function can be viewed as a ratio of the failure density function and survival function, the derivation is as follows,

$$\begin{aligned}
h_T(t|\boldsymbol{\theta}) &= \lim_{\delta t \to 0} \frac{\Pr(t < T \leq t + \delta t | T > t)}{\delta t} && \text{(2.3a)}\\
&= \lim_{\delta t \to 0} \frac{\Pr(t < T \leq t + \delta t, T > t)}{\Pr(T > t)\delta t} \\
&= \lim_{\delta t \to 0} \frac{\Pr(t < T \leq t + \delta t)}{\Pr(T > t)\delta t} && \Pr((A,B),A) = \Pr(A,B) \\
&= \lim_{\delta t \to 0} \frac{\Pr(t < T \leq t + \delta t)}{S_T(t|\boldsymbol{\theta})\delta t} \\
&= \frac{f_T(t|\boldsymbol{\theta})}{S_T(t|\boldsymbol{\theta})}. && \text{(2.3b)}
\end{aligned}$$

Clearly the hazard function is always positive, since it is a ratio of two positive functions. The hazard function can also be rewritten, using (2.1a), (2.2) and (2.3b), in the following ways,

$$\begin{aligned}
h_T(t|\boldsymbol{\theta}) &= \frac{-S'_T(t|\boldsymbol{\theta})}{S_T(t|\boldsymbol{\theta})} \\
&= -\frac{d}{dt} \log S_T(t|\boldsymbol{\theta}),
\end{aligned}$$

The *cumulative hazard function*, denoted $\Lambda_T(t|\boldsymbol{\theta})$, is the integral of all of the hazard up until time t,

$$\begin{aligned}\Lambda_T(t|\boldsymbol{\theta}) &= \int_0^t h_T(k|\boldsymbol{\theta})dk \qquad (2.4)\\ &= \left[\int h_T(k|\boldsymbol{\theta})dk\right]_0^t \\ &= \left[-\log S_T(k|\boldsymbol{\theta})\right]_0^t \\ &= -\log S_T(t|\boldsymbol{\theta}) - \log 1 \\ &= -\log S_T(t|\boldsymbol{\theta}), \qquad (2.5)\end{aligned}$$

The survival function can then be written in terms of the cumulative hazard function,

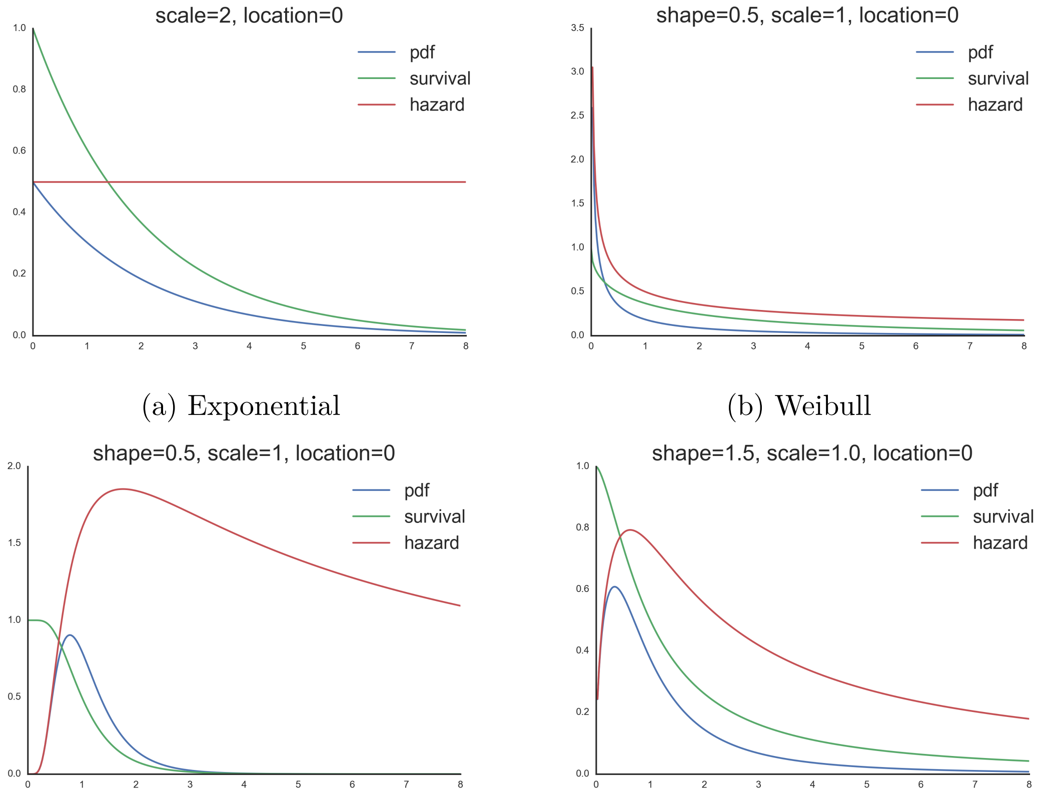
$$S_T(t|\boldsymbol{\theta}) = \exp(-\Lambda_T(t|\boldsymbol{\theta})), \qquad (2.6)$$

as can the failure density function,

$$\begin{aligned}f_T(t|\boldsymbol{\theta}) &= h_T(t|\boldsymbol{\theta})S_T(t|\boldsymbol{\theta}) \qquad (2.7)\\ &= h_T(t|\boldsymbol{\theta})\exp(-\Lambda_T(t|\boldsymbol{\theta})) \\ &= h_T(t|\boldsymbol{\theta})\exp\left(-\int_0^t h_T(k|\boldsymbol{\theta})dk\right). \qquad (2.8)\end{aligned}$$

The survival function, and hazard function can both be written in terms of the failure density function, Equation (2.1b) and Equation (2.3b) respectively. This means that as the shape of the failure density function is varied, or the distribution is changed, the survival function and hazard function also vary accordingly.

Figure 2.2 shows the effect different choices of failure density function has on the hazard and survival for a range of commonly used failure density functions. The cause of the failures may require different failure density functions, and correspondingly hazard functions. For example in the case of survival following surgery, it may be expected that the hazard of failure will start at its highest point, and then steadily reduce as the successful outcome of the surgery becomes more evident, a Weibull failure density function is capable of encoding this assumption with a certain choice of parameters, Figure 2.2b. Contrastingly, following onset of a disease such as cancer the hazard may start low, and rise, before falling, a property of the hazard that is more easily accommodated by a log-logistic distribution, Figure 2.2d. Different parameters of the chosen distribution can also result in different shapes of the distribution, for example with a different set of parameters, the log-logistic distribution can also accommodate



(a) Exponential

(b) Weibull

(c) Log-Normal

(d) Log-Logistic

Fig. 2.2 Visual indication of the relationship between the failure density function (pdf), survival function, and hazard function for a variety of different common distributions.

a monotonic decreasing hazard rate. This property will prove useful in Chapter 4 where we consider modelling multiple parameters of the failure time distribution with functions of observed covariates.

The main interests of survival analysis lie in inferring the distribution of survival times, analysing the influence of explanatory variables (covariates) on these distributions (Wan. Tang, 2012), and predicting lifetimes for other individuals based on these influences.

Cox, David and Oakes (1984) states a number of reasons consideration of the hazard function may be preferable to consideration of the failure density function including:

i It can be physically enlightening to consider the immediate 'risk' the individual has of observing the failure at the current time.

ii Comparisons between groups of individuals can sometimes be easier when considering hazards (see Section 2.4).

iii Convenient introduction of censoring times.

However, models can of course also be parameterised in terms of the failure density function, examples of both cases will be shown in the following sections.

2.1.5 Likelihood

The likelihood of a failure occurring can be constructed using the following argument. Consider n individuals that either die or are censored at times, $\mathbf{t} \in \mathbb{R}_{\geq 0}^{n \times 1}$, with failure indicators $\boldsymbol{\nu} \in \mathbb{Z}_{\{0,1\}}^{n \times 1}$ respectively. Assume that these times are independent and identically distributed (i.i.d) and drawn from a failure density function, $f_T(t|\boldsymbol{\theta})$, given the correct parameters for the density function that in many cases depend on covariate information of the individual. Their lifetimes are governed by a survival function $S_T(t|\boldsymbol{\theta})$, that is associated with a failure density function, $f_T(t|\boldsymbol{\theta})$, and hazard function $h_T(t|\boldsymbol{\theta})$ through the relationship in Equation (2.7). If the individual, i, fails at time t_i (not a censored observation) their contribution to the likelihood is,

$$L_i = f_T(t_i|\boldsymbol{\theta}) = S_T(t_i|\boldsymbol{\theta}) h_T(t_i|\boldsymbol{\theta}),$$

since it is known they survived up until t_i, the definition of $S_T(t_i|\boldsymbol{\theta})$, but then observed failure at the following instant, the definition of $h_T(t_i|\boldsymbol{\theta})$. If it is known the individual was alive up until time t_i, and must have survived beyond that time but there is no further information as to the exact time failure would be observed (the observation is right censored), their likelihood contribution is,

$$L_i = S_T(t_i|\boldsymbol{\theta}).$$

The full likelihood can then be written as either as a function of survival function and failure density function, or as a function of the survival function and hazard function,

$$\begin{aligned}
L &= \prod_{i=1}^{K:\nu=1} f_T(t_i|\boldsymbol{\theta}) \prod_{j=1}^{M:\nu=0} S_T(t_j|\boldsymbol{\theta}) && (2.9)\\
&= \prod_{i=1}^{N} f_T(t_i|\boldsymbol{\theta})^{\nu_i} S_T(t_i|\boldsymbol{\theta})^{1-\nu_i} && (2.10)\\
&= \prod_{i=1}^{N} (S_T(t_i|\boldsymbol{\theta}) h_T(t_i|\boldsymbol{\theta}))^{\nu_i} S_T(t_i|\boldsymbol{\theta})^{1-\nu_i}\\
&= \prod_{i=1}^{N} h_T(t_i|\boldsymbol{\theta})^{\nu_i} S_T(t_i|\boldsymbol{\theta})^{\nu_i} S_T(t_i|\boldsymbol{\theta})^{-\nu_i} S_T(t_i|\boldsymbol{\theta})\\
&= \prod_{i=1}^{N} h_T(t_i|\boldsymbol{\theta})^{\nu_i} S_T(t_i|\boldsymbol{\theta}), && (2.11)
\end{aligned}$$

where K is the set of all individuals where the event was observed (uncensored), and M is the set of all individuals where the event was not observed (censored).

The corresponding log likelihood, that will often be subsequently optimised, is as follows,

$$\log L = \sum_{i}^{N} \nu_i \log h_T(t_i|\boldsymbol{\theta}) - \Lambda_T(t_i|\boldsymbol{\theta}), \qquad (2.12)$$

where $\Lambda_T(t_i|\boldsymbol{\theta})$ indicates the cumulative hazard up until time t_i.

2.2 Kaplan Meier

Kaplan Meier (KM) is an estimator for a survival function that is used extensively in survival analysis, it is usually presented as a graphical plot of the survival observed by different groups in a clinical trial. One report (Pocock et al., 2007) indicated that over 40% of clinical trial analysis examined produce a Kaplan Meier estimate of their survival function. Its most attractive features are its simplicity, and its non-parametric nature requiring no parameters to be fit.

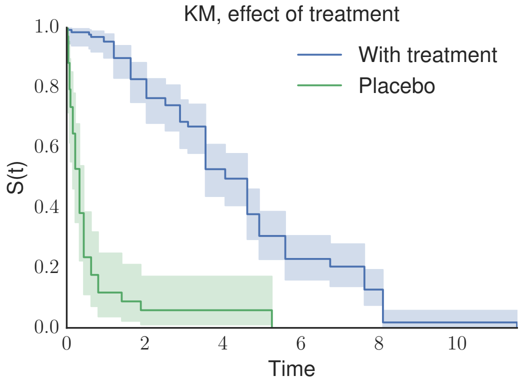


Fig. 2.3 Typical Kaplan-Meier plot with approximately proportional hazards

The probability of survival at any time using the KM estimator is calculated as follows,

$$\hat{S}(t) = \prod_{t_i < t} \frac{r_i - d_i}{r_i},$$

where $\hat{S}(t)$ is the approximate survival function. r_i is the number of individuals still at risk at the beginning of time t_i. d_i is the number of individuals that fail at time t_i. To compare the survival function between subgroups within a cohort, an approximate survival function is made for each subgroup, as illustrated in Figure 2.3. From visual inspection of the KM plot, it is clear that the subgroup receiving treatment has an increased expected median survival time.

Although KM estimators and plots are useful as an initial assessment of the data and the outcome of a clinical trial at a population or subgroup level, they have limitations. One such limitation is that individuals must be classified into discrete subgroups for comparison. Alongside the patient's failure times and clinical trial group, additional possibly continuous covariates for the patients may be available, that are believed to affect the individual's survival. Although it is possible to compare group survival using KM plots; to analyse multivariate data and continuous covariates, and subsequently make predictions for new individuals, more sophisticated survival analysis regression techniques are often required. There are a range of multivariate regression techniques that can produce an estimate of the effect each covariate may have on the failure time.

The two most popular approaches are the Cox proportional hazard model (Cox, David, 1972), Section 2.4, and the accelerated failure time class of models (Miller, 1976), Section 2.5. To unify these approaches alongside novel methods provided in the subsequent chapters, it will prove useful to recap on the definition of the class of generalised linear models.

2.3 Generalised Linear Modelling

Practical inference problems can often be posed as regression tasks. A popular approach to modelling when provided with a regression problem is to use a class of models known as *generalised linear models* (GLM) (McCullagh and Nelder, 1989; Nelder and Wedderburn, 1972).

Consider the case where a number of input-output pairings are provided by some system, $\{(\mathbf{x}_i, \mathbf{y}_i), i = 1, \ldots, n\}$, where $\mathbf{x}_i \in \mathbb{R}^{q\times 1}$ is regarded as an input in a q dimensional input space, and $\mathbf{y}_i \in \mathbb{R}^{p\times 1}$ is the corresponding observed output in a p dimensional output space. The full input matrix will be denoted $\mathbf{X} \in \mathbb{R}^{n\times q}$, and the full output matrix composing of p outputs will be denoted $\mathbf{Y} \in \mathbb{R}^{n\times p}$. In the survival analysis setting, inputs $\mathbf{X}$ may refer to the results of a series of q measurements being associated with n patients, known as covariates. Outputs, $\mathbf{Y}$, are observations that are deemed to be a *response* to some function of the covariates; for survival analysis failure time, t, is usually assumed to be a response, and so one of the p dimensions of $\mathbf{Y}$ will typically be survival times.

A classical linear regression model is written in the following way:

$$y_{i,j} = g_j(\mathbf{x}_{i,:}) + \epsilon_{i,j} \tag{2.13}$$

where $g_j(\mathbf{x}_{i,:}) = \boldsymbol{w}_{:,j}^{\top}\mathbf{x}_{i,:}$ and so assumes a linear association between the response variable y_i, and the explanatory variables, $\mathbf{x}_{i,:}$. $\boldsymbol{w}_{:,j}$ is a vector $\mathbb{R}^{q\times 1}$ that weights each explanatory variable, or covariate, accordingly for the output j. $\epsilon_{i,j}$ is known as the *noise model*, this component introduces an additional assumption that the observation we receive, $y_{i,j}$, is actually a *random* corruption of the linear component, $g_j(\mathbf{x}_{i,:})$. In standard linear regression $\epsilon_{i,j}$ is typically assumed to come from a Gaussian corruption, centred around the linear response, $\epsilon_{i,j} \sim \mathcal{N}(0, \beta^{-1})$.

Generalised linear models provide a framework that generalises the classical linear regression model in a number of ways. In practical problems the response variable $y_{i,j}$ often has some restrictions. An example that is very close to the heart of this thesis is the restriction that the observed time of failure must be non-negative, $t_i \in \mathbb{R}_{\geq 0}$. Another example would be that counts are typically discrete and non-negative, $y_{i,j} \in \mathbb{Z}_{\geq 0}$. By assuming a Gaussian noise model, this restriction cannot be fulfilled.

In generalised linear modelling parlance, in Equation (2.13) the observation $y_{i,j}$ is a normally distributed response variable, where the mean parameter of the distribution,

$\mu_{i,j}$, is linked to the explanatory variables, $\mathbf{x}_{i,:}$, through a *link function*, λ,

$$\lambda(\mu_{i,j}) = g_j(\mathbf{x}_{i,:}).$$

For the classical linear regression, Equation 2.13 the link function is the identity, $\lambda(a) = a$, and so in combination with the Gaussian noise model, $p(y_{i,j}|\mathbf{x}_{i,:}) = \mathcal{N}(y_{i,j}|\mu_{i,j}, \beta^{-1})$.

Under the GLM paradigm, the distribution is not restricted to be Gaussian, and may be any exponential family distribution (McCullagh and Nelder, 1989). Additionally the link function can be any *invertible* function. The inverse of the link function, $\lambda^{-1}(a)$, is known as the inverse link, or *transformation function*,

$$\mu_{i,j} = \lambda^{-1}(g_j(\mathbf{x}_{i,:}))$$

A common link function that will be used widely throughout this work is the *log link* function, $\lambda(a) = \log a$, corresponding to an exponential transformation function, $\lambda^{-1}(a) = \exp(a)$. By using the log link function, the parameter of the distribution is restricted to be a positive function of the explanatory variables, as the transformation function maintains $\lambda^{-1}(g_j(\mathbf{x}_{i,:})) \in \mathbb{R}_{>0}$.

The GLM paradigm maintains the restriction of the classical linear model that $g_j(\mathbf{x}_{i,:}) = \boldsymbol{w}_{:,j}^{\top}\mathbf{x}_{i,:}$. The generalised additive model (Hastie and Tibshirani, 1990) (GAM) relaxes the linear assumption of $g_j(\mathbf{x}_{i,:})$, by replacing it by a sum of smooth functions of the covariates x_i, giving

$$g_j(\mathbf{x}_{i,:}) = w_0 + \upsilon_1(x_1) + \upsilon_2(x_2) + \cdots,$$

where the smoothing functions $\upsilon_1, \upsilon_2, \ldots$, are left unspecified and learnt from the data. GAMs have also been applied to a class of survival analysis methods known as proportional hazards models (Lin and Ying, 1995). In Chapter 4 we will further extend this non-linear approach to survival analysis by using Bayesian non-parametric priors over functions.

2.4 Cox Proportional Hazards Model

The *Cox proportional hazards model* (CPH) or Cox regression is a multivariate regression technique for uncovering the effect of covariates on survival times. Perhaps the most widely used method in multivariate survival analysis, the model has been extremely popular with statisticians analysing the outcomes of clinical trials since its introduction in 1972 (Cox, David, 1972).

The model makes a *proportional hazards* assumption. This states that the proportion of two hazard functions for two individuals (see Section 2.1.4) is constant with respect to time, t. The overall hazard rate, $h_T(t|\boldsymbol{\theta}, \mathbf{x}_{i,:})$, for an individual i, now depends the individuals covariates $\mathbf{x}_{i,:}$ and is assumed to separate into a *baseline hazard rate*, $h_{0T}(t|\boldsymbol{\theta})$, and a *relative hazard rate*, $h_R(\boldsymbol{\theta}, \mathbf{x}_{i,:})$. The baseline hazard rate depends only on the time, t, as well as a set of parameters, $\boldsymbol{\theta}$. The relative hazard rate depends only on the covariates for the individual, $\mathbf{x}_{i,:}$ and some set of parameters, $\boldsymbol{\theta}$. In the simplest case the relative hazard is completely independent of time t, though the model has been extended for the case to handle time dependent covariates (Fisher and Lin, 1999).

The CPH model makes an assumption that all subjects *share* an underlying baseline hazard, that is then multiplicatively affected by the subject specific relative hazard. The overall hazard function for an individual i then factorise as

$$h_T(t|\boldsymbol{\theta}, \mathbf{x}_{i,:}) = h_{0T}(t|\boldsymbol{\theta})h_R(\boldsymbol{\theta}, \mathbf{x}_{i,:}).$$

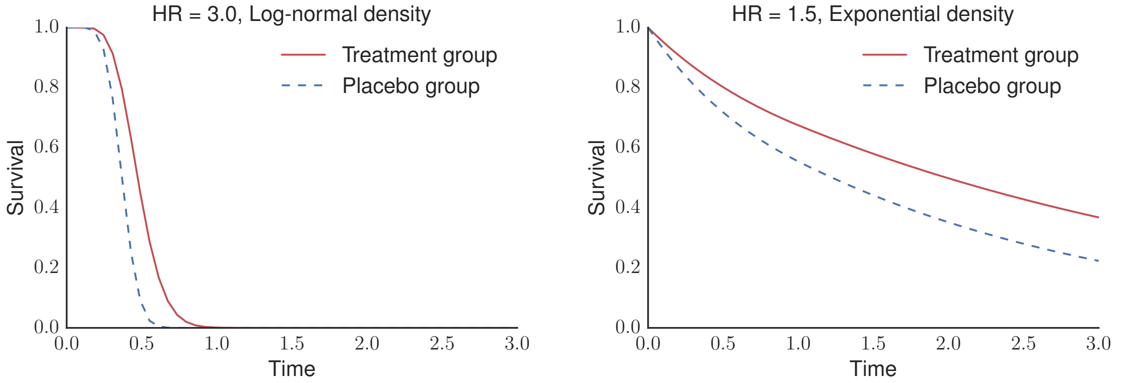
The baseline hazard function is defined in the following way, $h_{0T}(t|\boldsymbol{\theta}) \triangleq h_T(t|\boldsymbol{\theta}, \mathbf{x} = \mathbf{0})$; it describes the behaviour of the hazard function in the absence of any covariates, or of a 'baseline' individual, with all covariates equal to zero.

An important restriction in the context of this thesis is that the CPH model makes a log-linear assumption for the relative hazard, $h_R(\boldsymbol{\theta}, \mathbf{x}_{i,:}) = \exp(\boldsymbol{w}^\top \mathbf{x}_{i,:})$. The log-linear assumption constrains the relative hazard to be a positive quantity. This is a necessity as the overall hazard must be non-negative. In the context of GLMs a log-link function is used, such that $\log h_R(\boldsymbol{\theta}, \mathbf{x}_{i,:}) = \boldsymbol{w}^\top \mathbf{x}_{i,:}$. $\boldsymbol{\theta}$ includes a vector of weights, $\boldsymbol{w}$, as in GLMs; in this case each weight corresponds to the effect of a particular covariate measurement, encoding whether the covariates assist or hinder survival.

The model is often referred to as *semi-parametric* as it makes parametric assumptions about the effects of the covariates on the hazard function, but not about the shape of the baseline hazard itself (Royston and Lambert, 2011).

Since no form of the baseline hazard is assumed, failure time predictions cannot be made without first estimating the baseline hazard (Breslow, 1972). In practical implementations if it is needed, it is usually estimated in a separate step (Kalbfleisch and Prentice, 2002). These estimates give some indication of what the hazard function and survival function may look like, however it is not typically a smooth estimate.

Since the baseline hazard function is not modelled, the goal of the CPH model is not to make predictions of survival time for new individuals. The goal is instead to



(a) Hazard ratio of 3, low variance in T

(b) Hazard ratio of 1.5, high variance in T

Fig. 2.4 Larger hazard ratios (HR) do not show a percentage increase in time before the failure is expected. In the above figure, although the HR of 3 is much larger for Figure 2.4a than a HR of 1.5 in Figure 2.4b, clearly the benefits to overall survival time are much less significant in the former than the latter. The reason for such a difference is the presence of a different underlying survival distribution governing survival times. Figure 2.4b is governed by a survival distribution where the failures observed have a much larger variance. Figure inspired by Spruance et al. (2004).

estimate the *hazard ratio* (HR). The HR describes how the instantaneous risk varies proportionally between people with different covariates, assuming a shared baseline hazard. It does not aim to describe the survival function itself. The HR can be used to describe the overall effect that different covariates are likely to be having on the survival time, for example whether a certain treatment is effective, relative to another.

The CPH model could be criticised as it can be difficult to explain the findings in an intuitive way. For example, a hazard ratio value of $\exp(0.2) = 1.22$ does not mean that a person will live 22% longer if they have a covariate value of 1 rather than 0[1]. It only indicates that the *instantaneous risk* of the failure occurring in the next time-point is 22% higher. The HR does not provide a means to make absolute predictions of failure times, only the relative rate at which the failure is expected to be observed; as the absolute time also depends on the form of the baseline hazard, which is not specified or learnt. Figure 2.4 attempts to illustrate this distinction. This subtle difference can prove difficult to explain to clinicians and there are reports (Spruance et al., 2004) indicating that hazard ratios are often misinterpreted. These misinterpretations can give the impression that the gains are larger (or smaller) than what the model is really indicating.

[1] Hazard ratio for CPH model is given by $e^{\boldsymbol{w}} = e^{0.2} = 1.22$ when $\exp(\boldsymbol{w}) > 0$

An appealing feature of the CPH model, is that no specification of the form of the baseline hazard is required to infer the hazard ratios. This property can be seen by the definition of the hazard ratio,

$$\begin{aligned} HR &= \frac{\cancel{h_{0T}(t|\boldsymbol{\theta})}\exp(\boldsymbol{w}^\top \mathbf{x}_i)}{\cancel{h_{0T}(t|\boldsymbol{\theta})}\exp(\boldsymbol{w}^\top \mathbf{x}_j)} \\ &= \frac{\exp(\boldsymbol{w}^\top \mathbf{x}_i)}{\exp(\boldsymbol{w}^\top \mathbf{x}_j)}. \end{aligned}$$

Inference for the parameters $\boldsymbol{w}$ is done by maximising what is known as the *partial likelihood*[2].

From the perspective of a GLM, the CPH model can be seen as a log-odds model, of the overall hazard and the baseline hazard,

$$\begin{aligned} h_T(t|\boldsymbol{\theta}, \mathbf{x}_{i,:}) &= h_{0T}(t|\boldsymbol{\theta})h_R(\boldsymbol{\theta}, \mathbf{x}_{i,:}) \\ &= h_{0T}(t|\boldsymbol{\theta})\exp(\boldsymbol{w}^\top \mathbf{x}_{i,:}) \\ \log\left(\frac{h_T(t|\boldsymbol{\theta}, \mathbf{x}_{i,:})}{h_{0T}(t|\boldsymbol{\theta})}\right) &= \boldsymbol{w}^\top \mathbf{x}_{i,:}. \end{aligned}$$

In this case the standard model does not introduce a random error term, though this can be introduced to account for confounding variables that are not observed.

2.4.1 Proportional Hazards Likelihood

It is possible to derive the corresponding failure density function for a proportional hazards model on which the CPH model is based (McCullagh and Nelder, 1989), using Equation (2.8),

$$\begin{aligned} f_T(t|\boldsymbol{\theta}, \mathbf{x}_{i,:}) &= h_T(t|\boldsymbol{\theta}, \mathbf{x}_{i,:})\exp\left(-\int_0^t h_T(k|\boldsymbol{\theta}, \mathbf{x}_{i,:})dk\right) \\ &= h_{0T}(t|\boldsymbol{\theta})h_R(\boldsymbol{\theta}, \mathbf{x}_{i,:})\exp\left(-\int_0^t h_{0T}(k|\boldsymbol{\theta})h_R(\boldsymbol{\theta}, \mathbf{x}_{i,:})dk\right) \\ &= h_{0T}(t|\boldsymbol{\theta})\exp\left(\log h_R(\boldsymbol{\theta}, \mathbf{x}_{i,:}) - h_R(\boldsymbol{\theta}, \mathbf{x}_{i,:})\Lambda_{0T}(t|\boldsymbol{\theta})\right) \end{aligned}$$

where $\Lambda_{0T}(t|\boldsymbol{\theta})$ is the cumulative of the baseline hazard, analogically to the cumulative hazard.

[2] Jiezhi (2009) provides an intuitive derivation of the partial likelihood; a more in depth treatment is provided by Cox, David (1975).

Similarly the likelihood for the failure times, $\mathbf{t}$, occurring as they did whilst taking into account of censoring, $\boldsymbol{\nu}$; can be derived using Equation (2.12),

$$\begin{aligned} L &= \sum_{i=1}^{n} (\nu_i \log h_T(t|\boldsymbol{\theta}, \mathbf{x}_{i,:})) - \Lambda_T(t_i|\boldsymbol{\theta}, \mathbf{x}_{i,:})) \\ &= \sum_{i=1}^{n} (\nu_i \log h_{0T}(t|\boldsymbol{\theta}) h_R(\boldsymbol{\theta}, \mathbf{x}_{i,:})) - h_R(\boldsymbol{\theta}, \mathbf{x}_{i,:}) \Lambda_{0T}(t_i|\boldsymbol{\theta}) \end{aligned}$$

This likelihood function will be used as a basis for a more powerful extensions (Joensuu et al., 2013, 2012; Martino et al., 2010) that relax the log-linear assumption, as reviewed in Section 4.1.1.

2.4.2 Limitations

The CPH model does not come without its limitations. The CPH model relies on a proportionality assumption; that over time the ratio between two individuals hazard functions remain constant. This assumption in many cases may be too strong and may not hold depending on the data being analysed. Fortunately, there are many methods that can check whether the proportionality assumption is likely to hold, including log-cumulative hazard plots, Schoenfeld-residuals, crossing in Kaplan-Meier plots, amongst others (Collett, 2015; Klein and Moeschberger, 2003).

The second limitation is that the CPH model makes a log-linear assumption about the relative hazard, $h_R(\boldsymbol{\theta}, \mathbf{x}) = \exp(\boldsymbol{w}^\top \mathbf{x})$. This assumption in many cases may be invalid, as the linear effect of a covariate, $\boldsymbol{w}_1$, might vary in a non-linear way in response to the quantity of a covariate, $\mathbf{x}_1$. This limitation is widely known and has been observed to be invalid in many real data cases, including non-linearity of the effect of body mass index on mortality in the context of air pollutants (Krewski et al., 2003). Addressing this limitation within the probabilistic modelling framework is one of the focuses of this work.

2.5 Accelerated Failure Time Models

The proportional hazard assumption, Section 2.4, assumes that the effect of the covariates is to reduce or increase the overall proportion of subjects who observe the failures through time (Patel et al., 2006). Moreover, it assumes that this hazard is proportional over time.

An alternative method that does not necessarily assume the proportionality of hazards, is the class of *accelerated failure time models* (AFT) (Miller, 1976). These methods assume that the effect of the covariates is to accelerate or retard the survival time directly (Kleinbaum and Klein, 2006). Regression AFT models act directly on the lifetime (Smith, 2002) using some function of the covariates. This results in the "acceleration of time" within the survival function, not through multiplicative changes to a hazard function as in the proportional hazard model.

The model no longer *necessarily* assumes that the hazards between individuals are proportional. The results are arguably more interpretable, providing inference of the expected prolongation or reduction of the median failure time; an ability the hazard ratio does not immediately provide. It is often used when the proportionality of hazards does not hold (Collett, 2015).

There are a number of ways in which an AFT model can be written. We will provide two ways that provide helpful intuitions. The first will show transparently the property that AFT models share, that expected survival times are accelerated or contracted based on covariate values. The second interpretation will relate the model to the generalised linear modelling framework outlined in Section 2.3. This will prove useful when relating the standard AFT models described here, with the novel extensions presented in Chapter 4.

Assume a function of the covariates, that must be positive and so is written in terms of a positive transformation function, Section 2.3, $\lambda^{-1}(g(\mathbf{x}_{i,:}))$.

The first interpretation is in terms of the survival function, Section 2.1.3. AFT models act directly on the failure time, t, as follows,

$$S_T(t|\boldsymbol{\theta}, \mathbf{x}_{i,:}) = S_{0T}(\lambda^{-1}(g(\mathbf{x}_{i,:}))t|\boldsymbol{\theta}), \tag{2.14}$$

given a baseline survival function (for an individual with all covariates equal to zero), $S_{0T}(t|\boldsymbol{\theta}) = \exp(-\int_0^t h_{0T}(k|\boldsymbol{\theta})dk)$. It should be clear that a model defined in this way acts to accelerate the lifetime of the individual (if $\lambda^{-1}(g(\mathbf{x}_{i,:})) > 1$) resulting in earlier expected failure time, or retard the lifetime of the individual (if $0 < \lambda^{-1}(g(\mathbf{x}_{i,:})) < 1$) resulting in a later expected failure time than the baseline survival.

A more general form can be given for the class of AFT models (Kleinbaum and Klein, 2006). The random failure time, T_i for an individual is given by,

$$T_i = \lambda^{-1}(g(\mathbf{x}_{i,:}))T_0 \tag{2.15}$$

where the baseline failure density distribution for T_0 follows some specified parametric form as in Section 2.1.2; such as a Weibull, log-logistic, log-normal, etc. This shows that the baseline failure time distribution is either squashed or stretched according to function $\lambda^{-1}(g(\mathbf{x}_{i,:}))$, similarly to Equation (2.14). Stretching or squashing the failure time distribution suggests the failure occurrence is likely to happen later or earlier respectively.

By modelling the logarithm of random failure time, $\log T_i$,

$$\begin{aligned} \log T_i &= \log \lambda^{-1}(g(\mathbf{x}_{i,:})) + \log T_0 \\ &= g(\mathbf{x}_{i,:}) + \epsilon \end{aligned} \tag{2.16}$$

where we have assumed the transformation function λ^{-1} was an exponent. Assigning ϵ a logistic distribution makes T_0 a log-logistic distribution in Equation (2.15). Similarly assigning a Gaussian distribution for ϵ, means T_0 follows a log-normal distribution. A log-Weibull distribution (also known as the Gumbel distribution) for ϵ, gives a Weibull distribution for T_0. Within the GLM paradigm, AFT models are just log-linear models, as a linear restriction is assumed for $g(\mathbf{x}_{i,:}) = \boldsymbol{w}^\top \mathbf{x}$.

2.6 Conclusion

In this chapter the core components of survival analysis have been reviewed for the benefit of the unfamiliar reader. By framing both the CPH model and AFT models in terms of generalised linear models, the linear assumption about the effect of the covariates, $\mathbf{x}_{i,:}$ on the failure time distribution or hazard function is made clear. In practice the linearity assumption $g(\mathbf{x}_{i,:}) = \boldsymbol{w}^\top \mathbf{x}_{i,:}$ is restrictive. In many cases some prior information is available about the form of this function, but the exact form is unknown. Bayesian inference provides a framework in which this prior information can be incorporated, and uncertainty about the resulting inferred function can be maintained. In the following chapter the core components of Gaussian process priors are reviewed, that will provide a means by which a non-parametric function can be inferred from the data.

3

Gaussian Process Models

This chapter will cover the core principles of the Gaussian process (GP) class of models, that will be used extensively in the following chapters in combination with the survival analysis techniques covered in the preceding chapter. Gaussian processes provide a means to encode prior assumptions about a latent (unknown) function of interest, typically a regression model between input covariates and an observable output. Gaussian processes are non-parametric in nature, in the sense that they are not part of the class of parametric methods. Parametric methods principally rely on a fixed number of parameters, and inference is performed to uncover estimates for these fixed parameters; at this point the data used for inference can be discarded with no change to the underlying model. The number of parameters used within non-parametric methods grows with the number of data considered. Each datum forms part of the underlying model; data therefore cannot be discarded after inference without affecting the model, as the underlying parameters are intrinsically linked to the data. In the case of Gaussian processes, the growing number of parameters are typically concealed and do not need to be found through optimisation; instead parameters on which many of the underlying parameters rely are found. Perhaps the most powerful property of Gaussian processes is their ability to model non-linear functions, $g(\mathbf{x}_{i,:})$, and to do so whilst maintaining a measure of plausibility, unlike generalised linear models.

3.1 Gaussian Process Regression

Gaussian process regression again assumes a regression model of the following form,

$$y_{i,j} = g_j(\mathbf{x}_{i,:}) + \epsilon_{i,j},$$

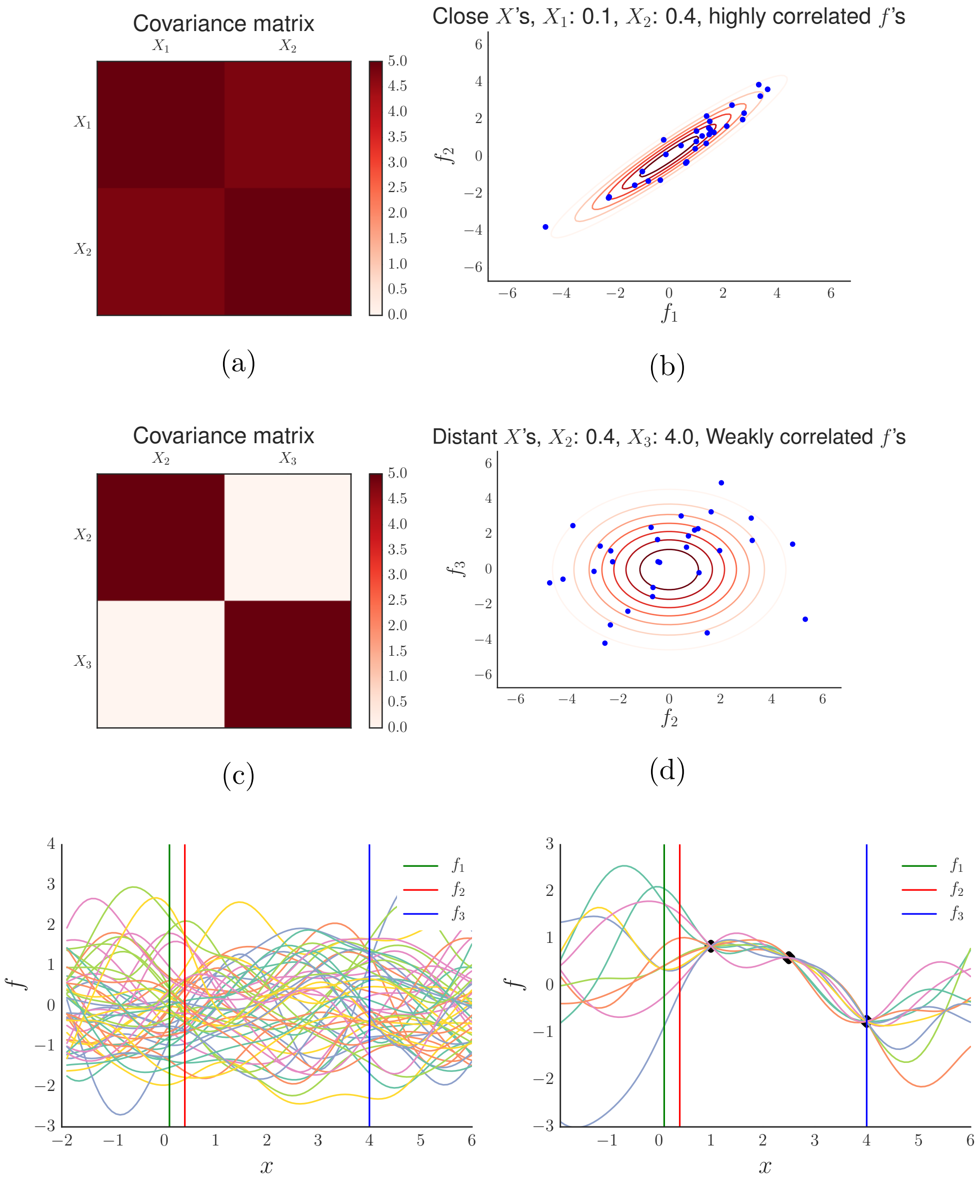
where in the simplest case $\epsilon_{i,j} \sim \mathcal{N}(0, \beta^{-1})$, and g_j denotes the function mapping the input to the jth output. However, unlike the classical linear regression model (Section 2.3), $g_j(\mathbf{x}_{i,:}) \triangleq f_{i,j}$, is assumed to come from an unobserved *random* latent function, realised at the location $\mathbf{x}_{i,:}$. This latent function is presumed to be an instance of a Gaussian process. To begin with we shall consider the case where the number of output dimensions, $p = 1$, and so $\mathbf{Y} \in \mathbb{R}^{n\times 1} \triangleq \mathbf{y}$. The case where multiple outputs are observed will be considered in Section 5.4.

Gaussian processes can be defined as a collection of random variables, where any finite number of which have a joint Gaussian distribution (Rasmussen and Williams, 2006). Intuitively they can also be seen to define a distribution over an infinite set of functions, where some functions are more probable than other functions; by sampling from this distribution possible functions can be generated.

Assumptions about how probable each of the infinite possible functions are, before seeing any data, is encoded in the Gaussian process *prior*, $p(\boldsymbol{f}|\mathbf{X})$. $\boldsymbol{f} \in \mathbb{R}^{n\times 1}$ denotes the random function values at locations $\mathbf{X} \in \mathbb{R}^{n\times q}$, where n is the number of input locations, and q is the number of dimensions of the input space. Since the random function values, $\boldsymbol{f}$, at any point are in fact a distribution of function values, a Gaussian process prior over the values is given as $p(\boldsymbol{f}|\mathbf{X}) = \mathcal{N}(\boldsymbol{f}|\mu(\mathbf{X}), k(\mathbf{X}, \mathbf{X}))$, which is characterized by a *kernel function*, $k(\mathbf{X}, \mathbf{X})$, and a *mean function*, $\mu(\mathbf{X})$. It is common to assume the mean function is zero, $\mu(\mathbf{X}) = \mathbf{0}$.

The kernel function computes the covariance matrix of the random variables for any finite set of inputs, $\mathbf{X}$. It also restricts covariance matrices that are generated to be positive semi-definite (Rasmussen and Williams, 2006) for any input of its required domain. For brevity $k(\mathbf{X}, \mathbf{X})$ is sometimes denoted $\mathbf{K}_{\boldsymbol{f}\boldsymbol{f}}$, since it denotes the covariance matrix between the n random function values $\boldsymbol{f}$, located at training inputs $\mathbf{X}$. Similarly $\mathbf{K}_{\boldsymbol{f}^*\boldsymbol{f}} \triangleq k(\mathbf{X}^*, \mathbf{X})$, denotes the covariance matrix between predictive outputs $\boldsymbol{f}^*$, located at $\mathbf{X}^*$, and the function $\boldsymbol{f}$ at training inputs $\mathbf{X}$. This notation is generalised for similar matrices, and will be used extensively throughout the remainder of this work. The kernel function typically depends on a number of hyper-parameters, $\boldsymbol{\theta}_K$, that are not always implicitly shown for notational simplicity.

The covariance matrix encodes how correlated the random function values, $\boldsymbol{f}$, are. The covariance matrix is generated by the kernel function based on how ‘similar’ the inputs $\mathbf{X}$ are, in some feature space. The feature space in which this similarity is computed varies between different choices of kernel functions (see Rasmussen and Williams (2006) for details of this interpretation). When inputs, $\mathbf{X}$, are similar to one another in the feature space the corresponding output random variables, $\boldsymbol{f}$, are more



(a)

(b)

(c)

(d)

(e) GP Prior draws

(f) GP posterior draws

Fig. 3.1 3.1a, 3.1b, 3.1c, 3.1d show the covariance matrix and joint samples for $\boldsymbol{f}_1$, $\boldsymbol{f}_2$, where $\mathbf{x}_1$ and $\mathbf{x}_2$ are similar and dissimilar in the feature space, producing highly correlated and weakly correlated function values, left and right respectively. 3.1e, 3.1f show draws from a GP prior and GP posterior respectively, with a RBF. Vertical lines show the locations of $\mathbf{x}_1$, $\mathbf{x}_2$ and $\mathbf{x}_3$.

highly correlated. Figure 3.1 illustrates the relationship between the kernel functions generated covariance matrix, and realisations of the random variables f_1 and f_2. If x_1 and x_2 are 'similar' in the feature space, the covariance is large (Figure 3.1a), and so the outputs f_1 and f_2 become highly correlated (Figure 3.1b). If x_1 and x_2 are 'dissimilar' from one another, the covariance is low (Figure 3.1c), and so the outputs f_1 and f_2 are weakly correlated (Figure 3.1d). More details of the implications of choosing a specific kernel functions are discussed in Section 3.1.1.

Depending on the latent function $\boldsymbol{f}$, evaluated at $\mathbf{X}$, the observed data, $\mathbf{y}$, may be probable or improbable under the model. This can be evaluated by computing the *likelihood*, $p(\mathbf{y}|\boldsymbol{f})$. The likelihood is usually chosen to follow a standard probability distribution. The *posterior* probability of function values, $p(\boldsymbol{f}|\mathbf{y}, \mathbf{X})$, can be inferred using Bayes rule, by combining the likelihood with the Gaussian process prior and dividing by the *marginal likelihood*, $p(\mathbf{y}|\mathbf{X})$.

If $\epsilon_i \sim \mathcal{N}\left(0, \beta^{-1}\right)$ and the data is assumed to be i.i.d, the likelihood of the observed data, $\mathbf{y}$, is can be written as follows,

$$\begin{aligned} p(\mathbf{y}|\boldsymbol{f}) &= \mathcal{N}\left(\mathbf{y}|\boldsymbol{f}, \beta^{-1}\mathbf{I}\right) && (3.1) \\ &= \prod_{i=1}^{n} \mathcal{N}\left(y_i|f_i, \beta^{-1}\right). && (3.2) \end{aligned}$$

Gaussian processes are frequently used in Bayesian statistics as a prior distribution over functions that may fit some data. Bayes rule allows us to combine our prior beliefs about the types of function that are expected with the likelihood of the data occurring. Our belief about the function having seen the data can then be updated,

$$\begin{aligned} p(\boldsymbol{f}|\mathbf{X}, \mathbf{y}) &= \frac{p(\boldsymbol{f}|\mathbf{X})p(\mathbf{y}|\boldsymbol{f})}{p(\mathbf{y}|\mathbf{X})} && (3.3) \\ &= \frac{p(\boldsymbol{f}|\mathbf{X})\prod_{i=1}^{n} p(y_i|f_i)}{p(\mathbf{y}|\mathbf{X})} && (3.4) \\ &= \frac{\mathcal{N}\left(\boldsymbol{f}|\mathbf{0}, k(\mathbf{X}, \mathbf{X})\right)\prod_{i=1}^{n} p(y_i|f_i)}{p(\mathbf{y}|\mathbf{X})} && (3.5) \\ &\propto \mathcal{N}\left(\boldsymbol{f}|\mathbf{0}, k(\mathbf{X}, \mathbf{X})\right)\prod_{i=1}^{n} \mathcal{N}\left(y_i|f_i, \beta^{-1}\right), && (3.6) \end{aligned}$$

where a i.i.d Gaussian likelihood assumption is specified as in Equation (3.2). By choosing such a Gaussian likelihood, it is assumed that each observation, y_i, is a Gaussian corruption of the underlying latent function value, f_i.

For many Bayesian inference problems the marginal likelihood of the observations, $p(\mathbf{y}|\mathbf{X}) = \int p(\mathbf{y}|\boldsymbol{f})p(\boldsymbol{f}|\mathbf{X})d\boldsymbol{f}$, is a tricky quantity to compute, and may require computationally expensive sampling methods. However, if the likelihood is assumed to be Gaussian as in Equation (3.2), and the prior, $p(\boldsymbol{f}|\mathbf{X})$, is a Gaussian process, then integration is analytical and results in another Gaussian distribution,

$$p(\mathbf{y}|\mathbf{X}) = \mathcal{N}\left(\mathbf{y}|\mathbf{0}, k(\mathbf{X},\mathbf{X}) + \beta^{-1}\mathbf{I}\right) \tag{3.7}$$

Note that the marginal covariance is simply the sum of the Gaussian prior covariance and the Gaussian likelihood covariance (see Appendix A.1, and Rasmussen and Williams (2006) for further useful properties of the Gaussian distribution).

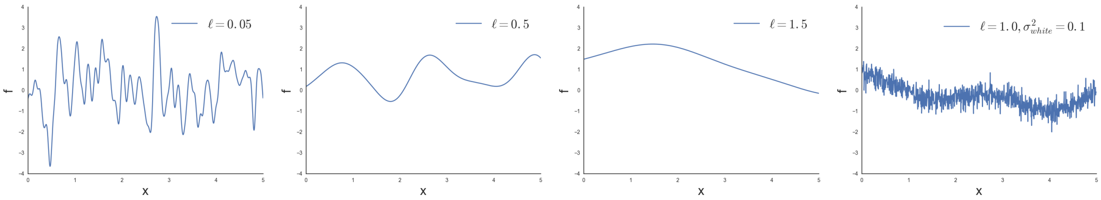
The ability to make predictions for new test points is also desirable. The posterior predictions for new test locations, denoted $\mathbf{X}^*$, can similarly be computed in closed form,

$$\begin{aligned} p(\boldsymbol{f}^*|\mathbf{X},\mathbf{y},\mathbf{X}^*) &= \mathcal{N}\left(\boldsymbol{f}^*|\boldsymbol{\mu}^*,\mathbf{K}^*\right) \\ \boldsymbol{\mu}^* &= k(\mathbf{X}^*,\mathbf{X})[k(\mathbf{X},\mathbf{X}) + \beta^{-1}\mathbf{I}]^{-1}\mathbf{y} \\ \mathbf{K}^* &= k(\mathbf{X}^*,\mathbf{X}^*) - k(\mathbf{X}^*,\mathbf{X})[k(\mathbf{X},\mathbf{X}) + \beta^{-1}\mathbf{I}]^{-1}k(\mathbf{X},\mathbf{X}^*) \end{aligned} \tag{3.8}$$

The posterior at the training points, $p(\boldsymbol{f}|\mathbf{X},\mathbf{y})$, is as in Equation (3.8), but with $\mathbf{X}^*$ replaced with $\mathbf{X}$.

If the likelihood is non-Gaussian, there is no such analytical solution. Handling this case will prove essential when Gaussian processes are being used for survival analysis in Chapter 4. When no analytical solution is available, approximations to the posterior must be made to make progress. A brief review of a number of methods for making posterior approximations, that will be built upon in subsequent chapters, is given in Section 3.2.

By combining the likelihood with the Gaussian process prior certain functions from the prior are deemed less or more likely depending on how well they fit the data and how probable they were under the prior. Figure 3.1f illustrates this by considering samples of the prior and the posterior of a Gaussian process. It is clear that functions from the prior that do not pass nearby the observations are deemed improbable, and unlikely to be drawn from the posterior. Figure 3.1e along with the earlier example given in Figure 3.1 illustrate how points (f_1 and f_2) are more highly correlated than points more distant (f_1 and f_3), in this case the ‘similarity’ is related to distance, by



(a) Small lengthscale

(b) Medium lengthscale

(c) Long lengthscale

(d) RBF + White kernel

Fig. 3.2 Draws from zero mean GPs with RBF kernel function with varying lengthscales, and addition of white noise kernel

assuming a kernel based on euclidean distance (in this case RBF as described in the next section).

3.1.1 Kernel Functions

A Gaussian process explicitly depends on the kernel function to compute the covariance between all the random variables of the latent function. The type of kernel chosen has an impact on the types of functions that are drawn from both the prior and the posterior.

Kernel functions are typically conditional on some hyper-parameters, $\boldsymbol{\theta}_K$, that further narrow down properties of the latent function. A *maximum a posteriori* (MAP) solution is typically found for these parameters, though it should be noted that this is an approximation that is only accurate when the posterior is strongly peaked. In Section 6.2 we consider a situation where this isn't true, the effect it can have, and consider some existing approaches to handling this issue.

An example of a very popular choice of kernel within the literature is the radial basis function kernel (RBF), also known as the exponentiated quadratic kernel, or Gaussian kernel. The RBF kernel requires two hyper-parameters, $\boldsymbol{\theta}_K = \{\ell, \sigma^2_{rbf}\}$, and is given by the following equation:

$$k(\mathbf{x}_{i,:}, \mathbf{x}_{j,:}) = \sigma^2_{rbf} \exp\left(-\frac{1}{2\ell^2} \sum_{k=1}^{q} (\mathbf{x}_{i,k} - \mathbf{x}_{j,k})^2\right).$$

A useful way of visualising the assumptions made by using a Gaussian process prior with a particular kernel, is to sample from the prior, $f \sim \mathcal{N}(\mathbf{0}, \mathbf{K}_{\boldsymbol{ff}})$, and plot it with respect to the input, $\mathbf{x}$. Figure 3.2 shows the effect that varying the lengthscale hyper-parameter, ℓ, has on samples from a Gaussian process priors with RBF kernels.

When the lengthscale is very large, draws of the Gaussian process prior are essentially linear within the input range. When the lengthscale is very small, each input, $\mathbf{x}_{i,:}$ is essentially independent of every other point $\mathbf{x}_{j,:}$, and appear like white noise. The RBF kernel however is infinitely differentiable (Rasmussen and Williams, 2006), and so functions drawn from a Gaussian process using the RBF kernel produce infinitely differentiable functions. If one was to zoom into the function using $\ell = 0.05$, the function maintains its smoothness.

Allowing the lengthscale to vary depending on the input dimension,

$$k(\mathbf{x}_{i,:}, \mathbf{x}_{j,:}) = \sigma^2_{rbf} \exp\left(-\frac{1}{2}\sum_{k=1}^{q}\frac{1}{\ell_k^2}(\mathbf{x}_{i,k} - \mathbf{x}_{j,k})^2\right),$$

gives the RBF automatic relevance determination (ARD) kernel (MacKay, 1996). In this case, when the kernel hyper-parameters are optimised, the model is able to learn that for some inputs $k \in [1, \cdots, q]$, the lengthscale can be very long compared to the scale of the data $\mathbf{x}_{:,k}$. It's often stated that this essentially suggests that input is irrelevant, since the covariance is essentially constant with respect to changes in either input, $\mathbf{x}_{i,k}$ or $\mathbf{x}_{j,k}$. In certain cases however it actually suggests that it is linear with respect to this input (Piironen and Vehtari, 2015), and is not necessarily irrelevant. If the lengthscale is smaller however, this indicates that covariance varies drastically depending on what the input values are, and these inputs are deemed relevant. Allowing the lengthscales to vary depending on the input dimension adds additional flexibility to the model.

The white noise kernel that assumes zero correlation between input locations can be written with the Kronecker delta function δ, a function that produces 1 when its two inputs are equal, and 0 otherwise,

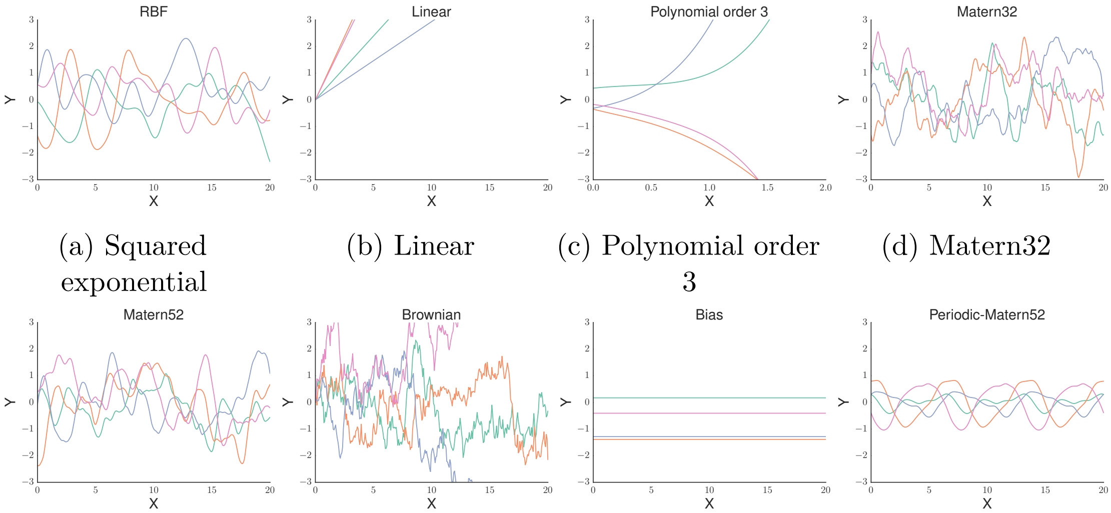
$$k(\mathbf{x}_{i,:}, \mathbf{x}_{j,:}) = \sigma^2_{white}\delta(i, j).$$

Another commonly used kernel function is the linear kernel function, that also does not depend on the in Euclidean distance between inputs $\mathbf{x}_{i,:} - \mathbf{x}_{j,:}$, is defined by the inner product,

$$k(\mathbf{x}_{i,:}, \mathbf{x}_{j,:}) = \sigma^2_{lin}\mathbf{x}_{i,:}^\top\mathbf{x}_{j,:}.$$

Gaussian process regression with a linear kernel transpires to be equivalent to Bayesian linear regression (Rasmussen and Williams, 2006).

Figure 3.3 shows draws from the prior of a range of different kernels encoding different assumptions about smoothness, linearity, differentiability, bias' and even



(a) Squared exponential

(b) Linear

(c) Polynomial order 3

(d) Matern32

(e) Matern52

(f) Brownian

(g) Bias

(h) Periodic Matern52

Fig. 3.3 Draws from zero mean GPs with a range of different kernel functions showing different assumptions about similarity between X locations. Equations for each kernel can be found in Appendix B or Section 3.1.1

periodicity. Kernel functions themselves can be combined in a multitude of ways, such as summation, multiplication and transformation of the input $\mathbf{x}$.

The summation of two kernels, results in the summation of the corresponding latent functions. This convenient property, amongst others, allows for complex prior distributions over functions to be defined trivially. For example, the RBF kernel assumes the true function is infinitely differentiable. In real data it is rarely likely that the function of interest is infinitely differentiable, and we may wish to encode this understanding within our prior over functions. It is common to relax this assumption by the addition of the white noise kernel, $k = k_{RBF} + k_{white}$. As can be seen in Figure 3.2d, the use of this summation kernel produces functions that are not smooth, but contain a smooth non-linear trend. This can be used as a tool to avoid problems of overfitting when the infinite differentiability assumption is likely to be violated in practice (Damianou, 2015).

The reader is directed towards Duvenaud (2014) for an intuitive visualisation and explanation of the implications of other methods of combining kernels.

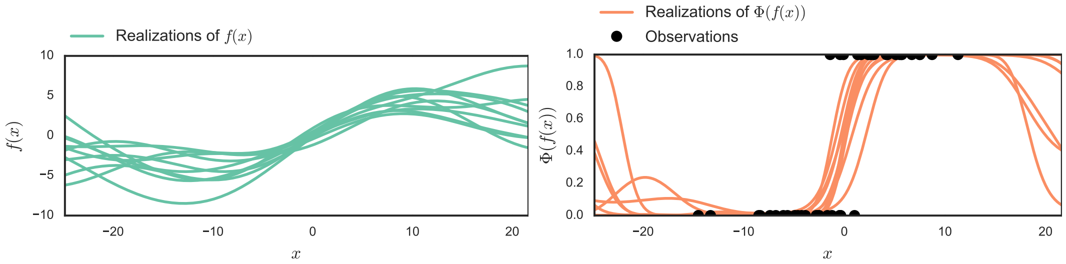


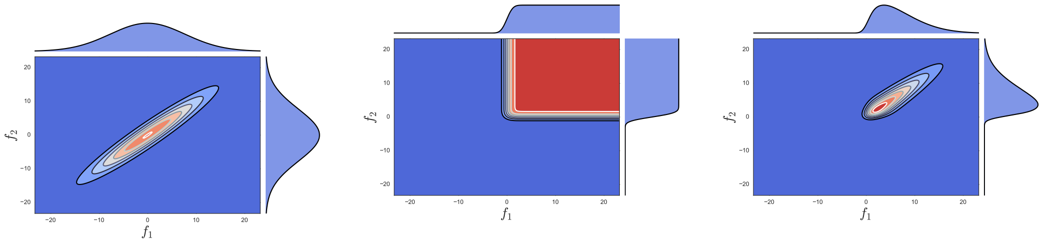
Fig. 3.4 Gaussian process classification, $\Phi(f(\mathbf{x}_{i,:}))$ can be used as the probability of the input $\mathbf{x}_{i,:}$ relating to a positive observation $y_i = 1$.

3.2 Approximations

Gaussian process regression has so far been considered in the context of a Gaussian likelihood, Equation (3.2). Since the Gaussian distribution is conjugate to itself, it is possible to obtain a closed form expression for the posterior $p(\boldsymbol{f}|\mathbf{y}, \mathbf{X})$, Equation (3.8), and marginal distribution, Equation (3.7). Assuming a Gaussian likelihood is however restrictive. The GLM framework reviewed in Section 2.3 can be used to relax this restriction for linear models, $g(\mathbf{x}_{i,:}) = \boldsymbol{w}^\top \mathbf{x}_{i,:}$. For example, a squashing function (probit), $\Phi(g(\mathbf{x}_{i,:}))$, can be specified as the inverse link function, ensuring $\pi_i = \Phi(g(\mathbf{x}_{i,:})) \in [0, 1]$. π_i in turn can be treated as a probability for the Bernoulli likelihood, $p(y_i|\pi_i) = \pi_i^{y_i}(1 - \pi_i)^{1-y_i}$, for binary observations $y_i \in \mathbb{Z}_{\{0,1\}}$.

An analogical case for Gaussian process classification, is illustrated in Figure 3.4, where the probit link function is used to squash the non-linear function, $\pi_i = \Phi(g(\mathbf{x}_{i,:})) = \Phi(\boldsymbol{f})$. The introduction of a non-Gaussian likelihood leads to a posterior $p(\boldsymbol{f}|\mathbf{y}, \mathbf{X})$ that is no longer analytically tractable, and it must be approximated. A number of methods for approximation have been proposed, including the Laplace approximation (Barber and Williams, 1997; Rue et al., 2009), variational approximations (Gibbs and MacKay, 2000; Jaakkola and Jordan, 1996; Nguyen and Bonilla, 2014; Opper and Archambeau, 2009; Seeger, 2000), Expectation Propagation (EP) (Hernández-Lobato and Hernández-Lobato, 2016; Minka, 2001; Seeger and Nickisch, 2011), Markov Chain Monte Carlo (MCMC) sampling (Neal, 1997), amongst others (Hensman et al., 2014).

Each of the above methods of approximation, with the exception of MCMC, approximate the non-Gaussian posterior, $p(\boldsymbol{f}|\mathbf{y})$ with a distribution, typically a Gaussian distribution, denoted $q(\boldsymbol{f})$. However, each approximation method differs in how the density of the true non-Gaussian posterior is captured by the Gaussian approximation. Figure 3.5 illustrates the true posterior of two latent points, f_1



(a) Prior $p(f_1, f_2)$

(b) Bernoulli likelihood $p(y = 1|f_1, f_2)$

(c) True posterior distribution $p(f_1, f_2|y = 1)$

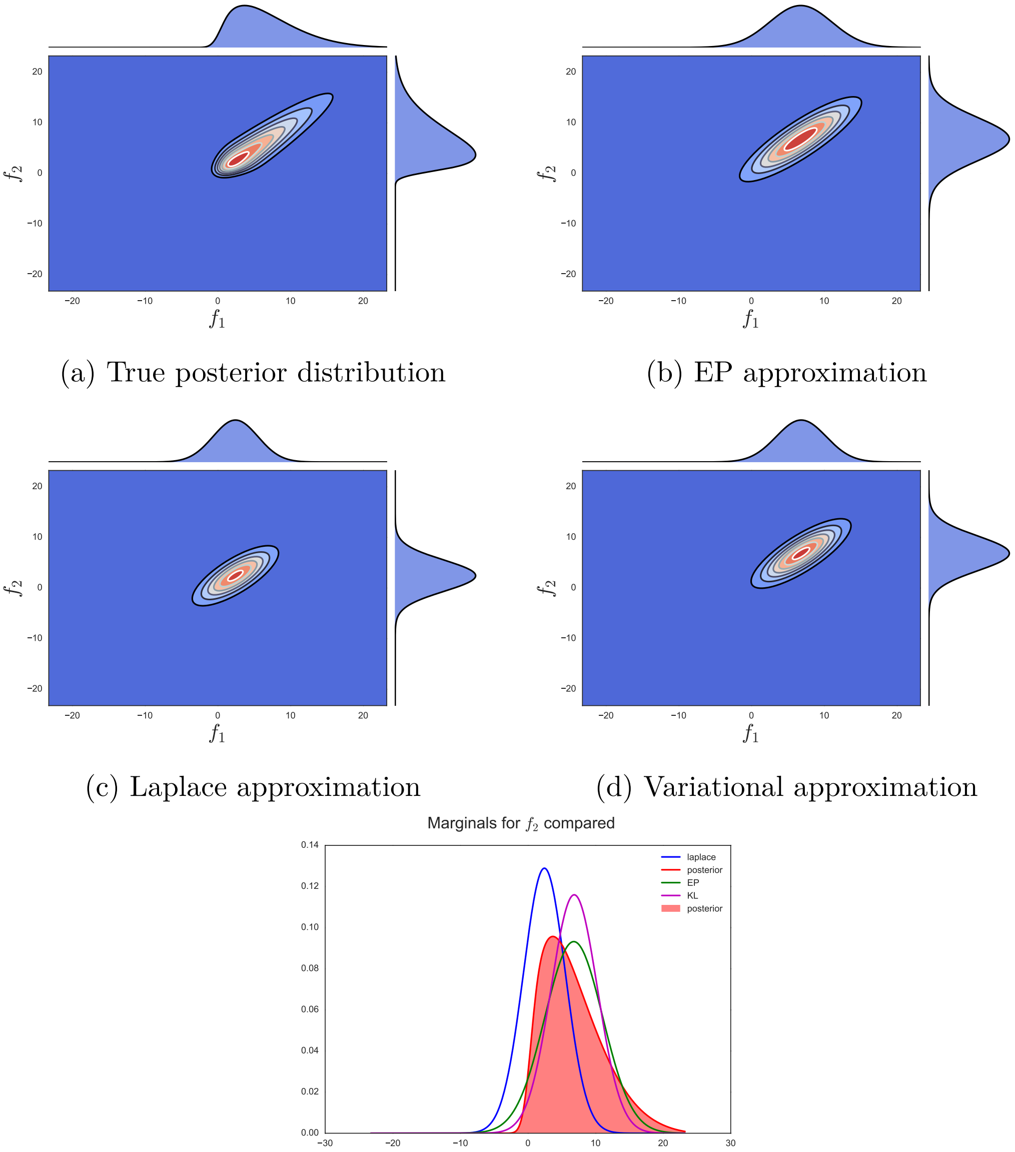
Fig. 3.5 Example showing how the combination of a Gaussian prior and Bernoulli likelihood gives rise to a non-Gaussian posterior distribution

and f_2, when a single observation, $y = 1$ is made, alongside a Gaussian prior and Bernoulli likelihood. The true posterior distribution is clearly non-Gaussian. Figure 3.6 illustrates the approximations that would be made by each Laplace, EP, and variational approximations.

A comparison of several approximate inference schemes for Gaussian process regression, with a Bernoulli likelihood specifically, was undertaken by Kuss and Rasmussen (2005) and Nickisch and Rasmussen (2008). In general EP is known to perform particularly well in this setting, but no single method consistently outperforms all others on the whole array of possible likelihood functions, each of the popular approximation methods are known to have a trade off between accuracy, simplicity and computational complexity.

This work will primarily focus on the Laplace approximation and variational approximation, and will consider a multitude of likelihood functions, including those that can be used for survival analysis. These methods will now be briefly reviewed. We choose not to focus on EP for a number of reasons. The method lacks guarantees of convergence, robust implementations can be difficult, and the Laplace approximation is known to perform well for survival likelihoods, matching the performance of EP in most cases [1]. MCMC has several advantages over both the Laplace approximation and the variational approximation, including exactness. However, it can be difficult to apply in many Gaussian process applications, in part because the many latent variables, $\boldsymbol{f}$, that need to be sampled can be very highly correlated (Titsias et al., 2008) which can be problematic for MCMC methods. To apply MCMC, uncorrelated samples must be taken, which may require many samples to be taken and so may incur a large computational overhead. Though both MCMC and EP have their difficulties, that is not to say that advances are not being made in these areas and it would be interesting

[1] Personal communication with Aki Vehtari (June 2016)



(a) True posterior distribution

(b) EP approximation

(c) Laplace approximation

(d) Variational approximation

(e) Approximations to the marginal of f_2

Fig. 3.6 Posterior approximations with differing methods of approximation, considering only two latent variables f_1 and f_2

to apply the ideas in this work considering both MCMC and EP; this is left to future work.

3.2.1 Laplace Approximation

The Laplace approximation, in the context of the Gaussian process framework, obtains a Gaussian approximation to the posterior, $p(\boldsymbol{f}|\mathbf{y})$. The method by which it forms this approximation is particularly simple, though care must be taken to maintain numerical stability (Rasmussen and Williams, 2006). The method begins by finding the mode of the true distribution, $p(\boldsymbol{f}|\mathbf{y})$. The curvature around the modal point is found and a Gaussian posterior approximation is made with a matching mode and curvature at this point. The results of this section will be used in the proceeding chapters in the context of survival analysis, and missing data imputation.

The posterior distribution can be written as a function,

$$p(\boldsymbol{f}|\mathbf{y}) = \frac{1}{Z} h(\boldsymbol{f}),$$

where $Z = p(\mathbf{y})$ is a normaliser that is constant in $\boldsymbol{f}$ and typically is difficult to compute. $h(\boldsymbol{f}) = p(\mathbf{y}|\boldsymbol{f})p(\boldsymbol{f})$ is the unnormalised posterior distribution. The Gaussian distribution has the property that its logarithm is a quadratic function of the variables (Bishop, 2006). Additionally a second order Taylor expansion of a function gives an approximation to the function up to a quadratic term, localised at a chosen point. Taking the logarithm of the posterior, $p(\boldsymbol{f}|\mathbf{y})$, and taking a second order Taylor expansion gives:

$$\begin{aligned}\log p(\boldsymbol{f}|\mathbf{y}) &= \log \frac{1}{Z} + \log h(\boldsymbol{f}) \\ &\approx \log \frac{1}{Z} + \log h(\mathbf{a}) + \frac{d \log h(\mathbf{a})}{d\mathbf{a}}(\boldsymbol{f} - \mathbf{a}) + \frac{1}{2}(\boldsymbol{f} - \mathbf{a})^\top \frac{d^2 \log h(\mathbf{a})}{d\mathbf{a}^2}(\boldsymbol{f} - \mathbf{a}).\end{aligned} \tag{3.9}$$

where $\mathbf{a}$ is the location at which the Taylor expansion is made. The Laplace approximation makes an approximation around the mode, denoted $\hat{\boldsymbol{f}}$, so

$$\left.\frac{d \log h(\mathbf{a})}{d\mathbf{a}}\right|_{\mathbf{a}=\hat{\boldsymbol{f}}} = \mathbf{0},$$

giving,

$$\log p(\boldsymbol{f}|\mathbf{y}) \approx \log \frac{1}{Z} + \log h(\hat{\boldsymbol{f}}) + \frac{1}{2}(\boldsymbol{f} - \hat{\boldsymbol{f}})^\top \frac{d^2 \log h(\hat{\boldsymbol{f}})}{d\hat{\boldsymbol{f}}^2}(\boldsymbol{f} - \hat{\boldsymbol{f}}),$$

taking exponents leaves

$$p(\boldsymbol{f}|\mathbf{y}) \approx \underbrace{\frac{1}{Z}h(\hat{\boldsymbol{f}})}_{\text{constant in } \boldsymbol{f}} \exp\left\{-\frac{1}{2}(\boldsymbol{f}-\hat{\boldsymbol{f}})^\top \left(-\frac{d^2 \log h(\hat{\boldsymbol{f}})}{d\hat{\boldsymbol{f}}^2}\right)(\boldsymbol{f}-\hat{\boldsymbol{f}})\right\}.$$

This is a exponentiated quadratic function of $\boldsymbol{f}$ and so can seen as an Gaussian distribution with an unknown normaliser, a mean given by $\hat{\boldsymbol{f}}$, and a precision matrix given by the negative Hessian,

$$\boldsymbol{A} = -\frac{d^2 \log h(\hat{\boldsymbol{f}})}{d\hat{\boldsymbol{f}}^2}.$$

Renormalising to find $\frac{1}{Z}$ leaves the Gaussian approximation to the posterior,

$$p(\boldsymbol{f}|\mathbf{y}) \approx q(\boldsymbol{f}) = \mathcal{N}\left(\boldsymbol{f}|\hat{\boldsymbol{f}}, \boldsymbol{A}^{-1}\right).$$

This approximation can be applied to Gaussian process regression with a non-Gaussian likelihood (Rasmussen and Williams, 2006). The key equations for applying the Laplace approximation in this context, assuming a Gaussian process prior for $p(\boldsymbol{f}) = \mathcal{N}(\boldsymbol{f}|\mathbf{0}, \mathbf{K}_{\boldsymbol{f}\boldsymbol{f}})$, are as follows,

$$\begin{aligned}
\log p(\boldsymbol{f}|\mathbf{y}) &= \log \frac{1}{p(\mathbf{y})} + \log p(\mathbf{y}|\boldsymbol{f}) + \log p(\boldsymbol{f}) \\
&\propto \log p(\mathbf{y}|\boldsymbol{f}) + \log p(\boldsymbol{f}) \\
&= \log p(\mathbf{y}|\boldsymbol{f}) - \frac{1}{2}\boldsymbol{f}^\top \mathbf{K}_{\boldsymbol{f}\boldsymbol{f}}^{-1}\boldsymbol{f} - \frac{1}{2}\log|\mathbf{K}_{\boldsymbol{f}\boldsymbol{f}}| - \frac{n}{2}\log 2\pi, \\
\frac{d \log p(\boldsymbol{f}|\mathbf{y})}{d\boldsymbol{f}} &= \frac{d \log p(\mathbf{y}|\boldsymbol{f})}{d\boldsymbol{f}} - \mathbf{K}_{\boldsymbol{f}\boldsymbol{f}}^{-1}\boldsymbol{f}, \\
\frac{d^2 \log p(\boldsymbol{f}|\mathbf{y})}{d\boldsymbol{f}^2} &= \frac{d^2 \log p(\mathbf{y}|\boldsymbol{f})}{d\boldsymbol{f}^2} - \mathbf{K}_{\boldsymbol{f}\boldsymbol{f}}^{-1} = -\boldsymbol{W} - \mathbf{K}_{\boldsymbol{f}\boldsymbol{f}}^{-1}, \qquad (3.10)
\end{aligned}$$

where $\boldsymbol{W} = -\frac{d^2 \log p(\mathbf{y}|\boldsymbol{f})}{d\boldsymbol{f}^2}$ as in Rasmussen and Williams (2006). Likelihoods that factorise, result in a diagonal matrix for $\boldsymbol{W}$, which is common to many likelihoods. Equation (3.10) gives the Hessian, and so $\boldsymbol{A} = \boldsymbol{W} + \mathbf{K}_{\boldsymbol{f}\boldsymbol{f}}^{-1}$ in Equation (3.2.1). However to evaluate the curvature, the modal point must first be found.

The modal point,

$$\frac{d \log p(\boldsymbol{f}|\mathbf{y})}{d\boldsymbol{f}} = \mathbf{0},$$

results in a self consistent equation for $\hat{\boldsymbol{f}}$,

$$\hat{\boldsymbol{f}} = \mathbf{K}_{\boldsymbol{ff}} \left(\frac{d \log p(\mathbf{y}|\hat{\boldsymbol{f}})}{d\hat{\boldsymbol{f}}} \right),$$

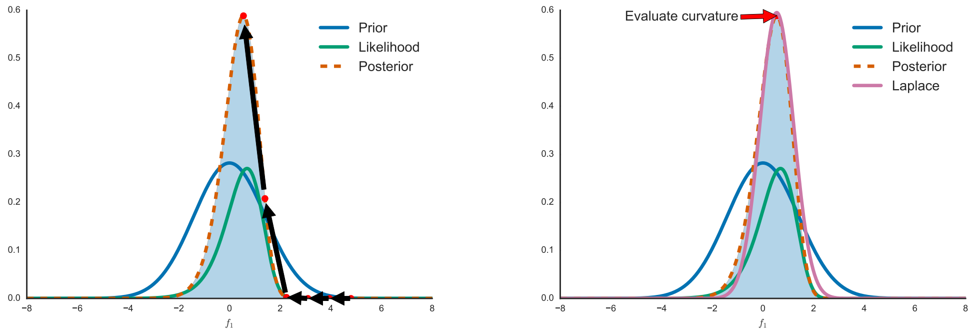
that due to non-linearities in $\frac{d \log p(\mathbf{y}|\hat{\boldsymbol{f}})}{d\hat{\boldsymbol{f}}}$, cannot be solved directly and must be solved iteratively, usually using Newton's method, see Rasmussen and Williams (2006) for details and implementation.

Using Equation (3.2.1) an approximation of the log marginal can be found easily, using the same Taylor expansion made in Equation (3.9),

$$\begin{aligned}
\log p(\mathbf{y}) &= \log \int p(\mathbf{y}|\boldsymbol{f})p(\boldsymbol{f})d\boldsymbol{f} \\
&\approx \log \int h(\hat{\boldsymbol{f}}) \exp(-\frac{1}{2}(\boldsymbol{f} - \hat{\boldsymbol{f}})^{\top} \boldsymbol{A}(\boldsymbol{f} - \hat{\boldsymbol{f}}))d\boldsymbol{f} \\
&= \log h(\hat{\boldsymbol{f}}) + \log \int \exp(-\frac{1}{2}(\boldsymbol{f} - \hat{\boldsymbol{f}})^{\top} \boldsymbol{A}(\boldsymbol{f} - \hat{\boldsymbol{f}}))d\boldsymbol{f} \\
&= \log p(\mathbf{y}|\hat{\boldsymbol{f}}) - \frac{1}{2}\hat{\boldsymbol{f}}^{\top}\mathbf{K}_{\boldsymbol{ff}}^{-1}\hat{\boldsymbol{f}} - \frac{1}{2}\log |\mathbf{K}_{\boldsymbol{ff}}|\,|\boldsymbol{A}| \\
&= \log p(\mathbf{y}|\hat{\boldsymbol{f}}) - \frac{1}{2}\hat{\boldsymbol{f}}^{\top}\mathbf{K}_{\boldsymbol{ff}}^{-1}\hat{\boldsymbol{f}} - \frac{1}{2}\log \left|\mathbf{I} + \boldsymbol{W}^{\frac{1}{2}}\mathbf{K}_{\boldsymbol{ff}}\boldsymbol{W}^{\frac{1}{2}}\right| \qquad (3.11)
\end{aligned}$$

where we have used the equality $\int \mathcal{N}\left(\boldsymbol{f}|\hat{\boldsymbol{f}}, \boldsymbol{A}^{-1}\right) d\boldsymbol{f} = 1$, for the unnormalised Gaussian in the second line. The final line uses the matrix determinant lemma (Appendix A.2.2) in reverse, and assumes that $\boldsymbol{W}$ can be factorised into $\boldsymbol{W}^{\frac{1}{2}}\boldsymbol{W}^{\frac{1}{2}}$, which is computationally inexpensive if $\boldsymbol{W}$ is diagonal, otherwise a LU decomposition may be used. $\left|\mathbf{I} + \boldsymbol{W}^{\frac{1}{2}}\mathbf{K}_{\boldsymbol{ff}}\boldsymbol{W}^{\frac{1}{2}}\right|$ will typically be more stable than $\left|\mathbf{K}_{\boldsymbol{ff}}^{-1}\right| |\mathbf{K}_{\boldsymbol{ff}} + \boldsymbol{W}|$ as pointed out by Rasmussen and Williams (2006).

The method by which a Laplace approximation is obtained is illustrated in Figure 3.7. Figure 3.6c depicts the Laplace approximation to the posterior generated with a Bernoulli likelihood, and a Gaussian prior over two variables f_1 and f_2. It shows that when the location and curvature at the modal point of a distribution is not descriptive of the distribution as a whole, the approximation can perform poorly. However, it has shown to be comparable to both EP and MCMC for a number of likelihood distributions (Riihimäki and Vehtari, 2014; Vanhatalo et al., 2010). We have found that it is effective for likelihoods that are used for survival analysis in this thesis.



(a) Mode finding via Newton optimisation (b) Evaluate curvature at mode, use mode and curvature to approximate the posterior

Fig. 3.7 Illustration of the Laplace approximation

3.2.2 Variational Approximation

There are a number of methods that have used variational inference in the context of Gaussian process inference for non-Gaussian likelihoods. This section will provide a recap of the method proposed by Opper and Archambeau (2009). Chapter 4 will build upon this method alongside the work of Lazaro-Gredilla and Titsias (2011) and Hensman et al. (2015b).

Readers unfamiliar with variational inference in general will find it useful to recap its key components, see Blei et al. (2016) for an excellent modern review.

Variational inference in general attempts to find a approximate distribution, $q(\boldsymbol{f}|\boldsymbol{\theta}_V)$, that closely matches a posterior distribution, $p(\boldsymbol{f}|\mathbf{y})$. The approximate distribution is conditioned on a set of *variational parameters*, $\boldsymbol{\theta}_V$, that only affect the quality of the approximation, not the distribution being approximated. It must belong to a family of tractable densities, where as in general the true posterior will not. The measure of closeness is the Kullback-Leibler (KL) divergence (Kullback, 1959; Kullback and Leibler, 1951). For continuous distributions, the KL divergence between two distributions, $q(\mathbf{x})$ and $p(\mathbf{x})$ is written

$$\mathrm{KL}\left(q(\mathbf{x})\,\|\,p(\mathbf{x})\right) = \int q(\mathbf{x}) \ln \frac{q(\mathbf{x})}{p(\mathbf{x})} d\mathbf{x}$$

The KL-divergence is an asymmetric measure, such that $\mathrm{KL}\left(q(\mathbf{x})\,\|\,p(\mathbf{x})\right) \neq \mathrm{KL}\left(p(\mathbf{x})\,\|\,q(\mathbf{x})\right)$. The KL-divergence is also always non-negative, $\mathrm{KL}\left(q(\mathbf{x})\,\|\,p(\mathbf{x})\right) \geq 0$. To make an approximation to the posterior for Gaussian process regression we can minimise a KL divergence between the approximate posterior, $q(\boldsymbol{f}|\boldsymbol{\theta}_V)$, and true posterior $p(\boldsymbol{f}|\mathbf{y})$. Variational inference uses the divergence $\mathrm{KL}\left(q(\boldsymbol{f}|\boldsymbol{\theta}_V)\,\|\,p(\boldsymbol{f}|\mathbf{y})\right)$.

EP (Minka, 2001) uses the reverse KL divergence at a local level to each random variable $\boldsymbol{f}$. Taking the expectation under the approximating distribution, $q(\boldsymbol{f}|\boldsymbol{\theta}_V)$, as in, $\mathrm{KL}\left(q(\boldsymbol{f}|\boldsymbol{\theta}_V) \,\|\, p(\boldsymbol{f}|\mathbf{y})\right)$, has the effect of encouraging the approximate distribution to have low density where the true distribution has low density. The reversed KL divergence, $\mathrm{KL}\left(p(\boldsymbol{f}|\mathbf{y}) \,\|\, q(\boldsymbol{f}|\boldsymbol{\theta}_V)\right)$, whereby the expectation is taken under the true distribution, $p(\boldsymbol{f}|\mathbf{y})$, has the effect of encouraging the approximation distribution to have a high density where the true distribution has a high density (Lawrence, 2000). Typically there will be a trade off between trying to capture density where there high density, and assigning density low where the true density is low. This effect is clearly visible comparing the approximations given by variational inference (Figure 3.6d) and EP (Figure 3.6b).

Opper and Archambeau (2009) introduced an approach to applying variational inference to Gaussian process regression. The basis of this variational approach has been expanded on in recent years (Hensman et al., 2015b; Khan et al., 2012; Lazaro-Gredilla and Titsias, 2011; Nguyen and Bonilla, 2014) and is used for the novel inference method put forward in Saul et al. (2016) that is covered in detail in Chapter 4. The method minimizes $\mathrm{KL}\left(q(\boldsymbol{f}|\boldsymbol{\theta}_V) \,\|\, p(\boldsymbol{f}|\mathbf{y})\right)$, in order to approximate the posterior distribution $p(\boldsymbol{f}|\mathbf{y}) = \frac{p(\mathbf{y},\boldsymbol{f})}{p(\mathbf{y})} = \frac{p(\mathbf{y}|\boldsymbol{f})p(\boldsymbol{f})}{p(\mathbf{y})}$, by exploiting the following equality,

$$
\begin{aligned}
\mathrm{KL}\left(q(\boldsymbol{f}|\boldsymbol{\theta}_V) \,\|\, p(\boldsymbol{f}|\mathbf{y})\right) &= \int q(\boldsymbol{f}|\boldsymbol{\theta}_V) \log \frac{q(\boldsymbol{f}|\boldsymbol{\theta}_V)}{p(\boldsymbol{f}|\mathbf{y})} d\boldsymbol{f} \\
&= \log p(\mathbf{y}) + \int q(\boldsymbol{f}|\boldsymbol{\theta}_V) \log \frac{q(\boldsymbol{f}|\boldsymbol{\theta}_V)}{p(\mathbf{y},\boldsymbol{f})} d\boldsymbol{f} \\
\log p(\mathbf{y}) &= -\int q(\boldsymbol{f}|\boldsymbol{\theta}_V) \log \frac{q(\boldsymbol{f}|\boldsymbol{\theta}_V)}{p(\mathbf{y},\boldsymbol{f})} d\boldsymbol{f} + \mathrm{KL}\left(q(\boldsymbol{f}|\boldsymbol{\theta}_V) \,\|\, p(\boldsymbol{f}|\mathbf{y})\right) \\
&= \int q(\boldsymbol{f}|\boldsymbol{\theta}_V) \log p(\mathbf{y}|\boldsymbol{f}) d\boldsymbol{f} - \mathrm{KL}\left(q(\boldsymbol{f}|\boldsymbol{\theta}_V) \,\|\, p(\boldsymbol{f})\right) \\
&\quad + \mathrm{KL}\left(q(\boldsymbol{f}|\boldsymbol{\theta}_V) \,\|\, p(\boldsymbol{f}|\mathbf{y})\right)
\end{aligned} \tag{3.12}
$$

In practice, since the true posterior, $p(\boldsymbol{f}|\mathbf{y})$, will not usually belong to a family of tractable distributions whilst the approximating distribution, $q(\boldsymbol{f}|\boldsymbol{\theta}_V)$, must, the KL divergence, $\mathrm{KL}\left(q(\boldsymbol{f}|\boldsymbol{\theta}_V) \,\|\, p(\boldsymbol{f}|\mathbf{y})\right)$, will not be computable. Using the fact that a KL-divergence is non-negative an inequality is obtained,

$$
\log p(\mathbf{y}) \geq \int q(\boldsymbol{f}|\boldsymbol{\theta}_V) \log p(\mathbf{y}|\boldsymbol{f}) d\boldsymbol{f} - \mathrm{KL}\left(q(\boldsymbol{f}|\boldsymbol{\theta}_V) \,\|\, p(\boldsymbol{f})\right). \tag{3.13}
$$

More generally this can be written as,

$$\log p(\mathbf{y}) \geq \int q(\boldsymbol{f}|\boldsymbol{\theta}_V)[\log p(\boldsymbol{f}, \mathbf{y}) - \log q(\boldsymbol{f}|\boldsymbol{\theta}_V)]d\boldsymbol{f}.$$

In variational inference this inequality is known as the evidence lower bound (ELBO), and is widely used (Hoffman et al., 2013; Nguyen and Bonilla, 2014; Ranganath et al., 2014) as it provides a lower bound on the true log evidence $\log p(\mathbf{y})$. Maximising the ELBO is equivalent to minimising $\mathrm{KL}\left(q(\boldsymbol{f}|\boldsymbol{\theta}_V) \,\|\, p(\boldsymbol{f}|\mathbf{y})\right)$ (Bishop, 2006; Jordan et al., 1999; Ranganath et al., 2014), which makes the approximate posterior as similar as possible to the true posterior. The first term inside the expectation can be seen to encourage parameters of the variational distribution that give high density to configurations of the latent variables that also explain the observations, $\mathbf{y}$. The second term encourages parameters that give rise to entropic variational distributions; such that the distribution spreads it mass across many configurations (Ranganath et al., 2014).

Suitable variational parameters, $\boldsymbol{\theta}_V$, can be found by maximising Equation (3.13). Opper and Archambeau (2009) assume the approximating distribution, $q(\boldsymbol{f}|\boldsymbol{\theta}_V)$, is a Gaussian distribution $q(\boldsymbol{f}|\boldsymbol{\theta}_V) = \mathcal{N}(\boldsymbol{f}|\boldsymbol{\mu}_V, \boldsymbol{\Sigma}_V)$, where the variational parameters are $\boldsymbol{\theta}_V = \{\boldsymbol{\mu}_V, \boldsymbol{\Sigma}_V\}$; for a factorising likelihood this gives the bound to be maximised as follows,

$$\begin{aligned} L(q) = &\frac{1}{2}\mathrm{Tr}(\mathbf{K}_{\boldsymbol{ff}}\boldsymbol{\Sigma}_V) + \frac{1}{2}\boldsymbol{\mu}_V{}^\top \mathbf{K}_{\boldsymbol{ff}}^{-1}\boldsymbol{\mu}_V - \frac{1}{2}\ln|\boldsymbol{\Sigma}_V| + \ln Z - \frac{n}{2}\ln 2\pi e \\ &- \sum_{i=1}^{n}\int q(\boldsymbol{f}|\boldsymbol{\theta}_V)\log p(\mathbf{y}_i|\boldsymbol{f}_i)d\boldsymbol{f}. \end{aligned}$$

Furthermore note that each $\int q(\boldsymbol{f}|\boldsymbol{\theta}_V)\log p(\mathbf{y}_i|\boldsymbol{f}_i)d\boldsymbol{f}$ only depends on the corresponding mean $\boldsymbol{\mu}_{V_i}$ and covariance element $\boldsymbol{\Sigma}_{V_{i,i}}$ of the full covariance matrix, $\boldsymbol{\Sigma}_V$. Using the equality $\boldsymbol{\Sigma}_V{}^{-1} = -2\nabla_{\boldsymbol{\Sigma}_V}\mathbb{E}_{q(\boldsymbol{f}|\boldsymbol{\theta}_V)}\Big[\log p(\mathbf{y}|\boldsymbol{f})\Big]$ allows the authors to express the optimal covariance as,

$$\boldsymbol{\Sigma}_V = (\mathbf{K}_{\boldsymbol{ff}}^{-1} + \boldsymbol{\Omega}_V)^{-1}$$

where $\boldsymbol{\Omega}_V$ is a diagonal matrix with variational parameters $\boldsymbol{\omega}_V \in \mathbb{R}^{n\times 1}$, along its diagonal, and so only $2n$ variational parameters are required. Without this trick, the naïve representation would require in total $n(n+3)/2$ variational parameters to be learnt. By applying this trick only $2n$ variational parameters are required. The variational parameters are optimised using a gradient decent method. By significantly reducing the number of parameters that must be optimised, local minima can be avoided resulting in a better approximations to the posterior.

3.3 Sparse Gaussian Process

Gaussian processes have proven to be popular methods in a myriad of application areas. This is a testament to their flexible modelling assumptions through the covariance function (Section 3.1.1), and their ability to quantify uncertainty in areas of the input space that are sparsely populated by data. In their standard form as presented in Section 3.1 however there is a significant computational complexity and memory overhead associated with them when applied to anything but modest dataset sizes. As data in many areas has become increasingly plentiful, this poses a practical problem. This complexity arises from the need to compute the inverse of the covariance matrix $\mathbf{K}_{\boldsymbol{ff}}$, which is a $n \times n$ matrix. The storage of the covariance matrix requires $\mathcal{O}(n^2)$ memory locations. The inversion itself requires a computational complexity of $\mathcal{O}(n^3)$. The practical limit of this method is $n \approx 10000$; after which such computations become excessively burdensome even on high performance machines.

A number of methods for reducing the computation and storage requirements have been put forward in the literature. The Bayesian committee machine (Tresp, 2000) and variants around the same idea (Cao and Fleet, 2014; Deisenroth and Ng, 2015), attempt to address the computational problem by training a number of "expert" models on subsets of the data, and then combine inferences in a consistent manner. By far the most popular approach to tackling the computational bottleneck however, come under the umbrella of *sparse approximations* (Quiñonero Candela and Rasmussen, 2005; Seeger et al., 2003; Snelson and Ghahramani, 2006a; Titsias, 2009). Sparse approximations seek to approximate the full covariance matrix, $\mathbf{K}_{\boldsymbol{ff}}$, with a low rank form. The methods approach this problem by introducing an auxiliary set of m input variables, $\mathbf{U} \in \mathbb{R}^{m \times p}$, known as *inducing points*; that live in the same space as $\boldsymbol{F} \in \mathbb{R}^{n \times p}$. Inducing points are hypothetical evaluations of the latent function $g(\mathbf{Z})$, where $\mathbf{Z} \in \mathbb{R}^{m \times q}$ are known as *inducing inputs*. Inducing inputs are typically either chosen as a subset of the training data $\mathbf{X}$, (Quiñonero Candela and Rasmussen, 2005) or optimised (Hensman et al., 2013; Titsias, 2009). For simplicity, we will again consider the case where $p = 1$, that is there is only one output dimension.

Since the inducing variables, $\mathbf{u}$ are additional realisations of the latent function $p(\boldsymbol{f}|\mathbf{X})$, but at locations $\mathbf{Z}$, their prior is $p(\mathbf{u}|\mathbf{Z}) = \mathcal{N}(\mathbf{u}|\mathbf{0}, \mathbf{K}_{\mathbf{uu}})$; analogically to $\mathbf{K}_{\boldsymbol{ff}} = k(\mathbf{X}, \mathbf{X})$, the notation $\mathbf{K}_{\mathbf{uu}} = k(\mathbf{Z}, \mathbf{Z})$ is used. The joint distribution is

Gaussian distribution,

$$p(\boldsymbol{f}, \mathbf{u}|\mathbf{X}, \mathbf{Z}) = \mathcal{N}\left(\begin{bmatrix}\boldsymbol{f}\\ \mathbf{u}\end{bmatrix} \mid \begin{bmatrix}0\\ 0\end{bmatrix}, \begin{bmatrix}\mathbf{K}_{\boldsymbol{f}\boldsymbol{f}} & \mathbf{K}_{\boldsymbol{f}\mathbf{u}}\\ \mathbf{K}_{\mathbf{u}\boldsymbol{f}} & \mathbf{K}_{\mathbf{u}\mathbf{u}}\end{bmatrix}\right),$$

where $\mathbf{K}_{\boldsymbol{f}\mathbf{u}} = k(\mathbf{X}, \mathbf{Z})$ and $\mathbf{K}_{\mathbf{u}\boldsymbol{f}} = k(\mathbf{Z}, \mathbf{X}) = \mathbf{K}_{\boldsymbol{f}\mathbf{u}}^{\top}$. Similar notation is used for predictive points, $\mathbf{u}^*$ and of course $\boldsymbol{f}^*$. Using the Gaussian identities, Appendix A.1, the conditional distribution $p(\boldsymbol{f}|\mathbf{u}, \mathbf{X})$ can be written

$$p(\boldsymbol{f}|\mathbf{u}, \mathbf{X}) = \mathcal{N}(\boldsymbol{f}|\mathbf{K}_{\boldsymbol{f}\mathbf{u}}\mathbf{K}_{\mathbf{u}\mathbf{u}}^{-1}\mathbf{u}, \underbrace{\mathbf{K}_{\boldsymbol{f}\boldsymbol{f}} - \mathbf{Q}_{\mathbf{ff}}}_{\tilde{\mathbf{K}}}) \tag{3.14}$$

$$\mathbf{Q}_{\mathbf{ff}} = \mathbf{K}_{\boldsymbol{f}\mathbf{u}}\mathbf{K}_{\mathbf{u}\mathbf{u}}^{-1}\mathbf{K}_{\mathbf{u}\boldsymbol{f}}. \tag{3.15}$$

Incorporating the likelihood $p(\mathbf{y}|\boldsymbol{f})$ results in the joint model,

$$\begin{aligned} p(\mathbf{y}, \boldsymbol{f}, \mathbf{u}|\mathbf{X}, \mathbf{Z}) &= p(\mathbf{y}|\boldsymbol{f})p(\boldsymbol{f}|\mathbf{u}, \mathbf{X})p(\mathbf{u}|\mathbf{Z}) \\ p(\mathbf{u}|\mathbf{Z}) &= \mathcal{N}\left(\mathbf{u}|\mathbf{0}, \mathbf{K}_{\mathbf{u}\mathbf{u}}\right) \\ p(\boldsymbol{f}|\mathbf{u}, \mathbf{X}) &= \mathcal{N}\left(\boldsymbol{f}|\mathbf{K}_{\boldsymbol{f}\mathbf{u}}\mathbf{K}_{\mathbf{u}\mathbf{u}}^{-1}\mathbf{u}, \tilde{\mathbf{K}}\right). \end{aligned}$$

Due to the properties of the Gaussian distribution, $\mathbf{u}$ can be marginalized resulting in no change from the existing Gaussian process regression models (Section 3.1) prior,

$$\begin{aligned} p(\boldsymbol{f}|\mathbf{X}) &= \int p(\boldsymbol{f}|\mathbf{u}, \mathbf{X})p(\mathbf{u}|\mathbf{Z})d\mathbf{u} \\ &= \mathcal{N}\left(\boldsymbol{f}|\mathbf{K}_{\boldsymbol{f}\mathbf{u}}\mathbf{K}_{\mathbf{u}\mathbf{u}}^{-1}\mathbf{0}, \mathbf{K}_{\boldsymbol{f}\boldsymbol{f}} - \mathbf{K}_{\boldsymbol{f}\mathbf{u}}\mathbf{K}_{\mathbf{u}\mathbf{u}}^{-1}\mathbf{K}_{\mathbf{u}\boldsymbol{f}} + \mathbf{K}_{\boldsymbol{f}\mathbf{u}}\mathbf{K}_{\mathbf{u}\mathbf{u}}^{-1}\mathbf{K}_{\mathbf{u}\boldsymbol{f}}\right) \\ &= \mathcal{N}\left(\boldsymbol{f}|\mathbf{0}, \mathbf{K}_{\boldsymbol{f}\boldsymbol{f}}\right), \end{aligned}$$

where the conditioning can be dropped on $\mathbf{Z}$. Notice that the in its current form, the addition of the inducing variables $\mathbf{u}$ does not alleviate the $\mathcal{O}(n^3)$ computational burden. To make progress sparse methods make an approximation, $q(\boldsymbol{f}|\mathbf{u}, \mathbf{X})$, to the conditional posterior, Equation (3.14). This is done by replacing $\tilde{\mathbf{K}}$ with an alternative covariance matrix, that induces similar covariances between random variables, $\boldsymbol{f}$, but that do not require an inversion of a $\mathcal{O}(n^3)$ matrix. Typically the computational complexity of sparse approximation methods are $\mathcal{O}(nm^2)$, and the storage requirements are $\mathcal{O}(nm)$. Computational and memory savings arise when m is chosen such that $m < n$.

A variety of suggestions have been made for how $\tilde{\mathbf{K}}$ should be approximated; the most prominent to date have been deterministic training conditional (DTC) (Csató and Opper, 2002; Seeger et al., 2003), fully independent training conditional (FITC) (Snelson and Ghahramani, 2005), and the variational sparse GP approach of Titsias (2009). A unifying view of the former two methods, alongside less popular alternatives was made by Quiñonero Candela and Rasmussen (2005). A review by Damianou (2015) attempts to integrate the more recent variational approach within this unified view.

Following is a recap of the essential components of the variational approach, that will be required to follow the novel aspects of this work.

3.3.1 Variational Sparse Gaussian Processes

The variational sparse Gaussian process approach (Titsias, 2009) like all variational inference methods, attempts to minimize a KL divergence. The KL divergence to be minimised in this case is $\mathrm{KL}\left(q(\boldsymbol{f}|\boldsymbol{\theta}_V) \,\|\, p(\boldsymbol{f},\mathbf{u}|\mathbf{y})\right)$, where a specific factorisation $q(\boldsymbol{f}|\boldsymbol{\theta}_V) = p(\boldsymbol{f}|\mathbf{u})q(\mathbf{u}|\boldsymbol{\theta}_V)$ is used for the approximating distribution, rather than the true posterior $p(\boldsymbol{f}|\mathbf{y},\mathbf{u})p(\mathbf{u}|\mathbf{y})$. This suggests that information about the observations, $\mathbf{y}$, in the true posterior $p(\boldsymbol{f},\mathbf{u}|\mathbf{y})$ must be compressed into the distribution that contains flexibility, $q(\mathbf{u}|\boldsymbol{\theta}_V)$. If $q(\mathbf{u}|\boldsymbol{\theta}_V)$ is sufficient for representing the information that would otherwise be encoded in $p(\boldsymbol{f},\mathbf{u}|\mathbf{y})$, the approximation will perform well; if this does not hold, for example if the true function is complex and requires all the information held at all n training data, it will perform poorly.

Following the standard variational inference framework, the following bound is found

$$
\begin{aligned}
\mathrm{KL}\left(q(\boldsymbol{f}|\boldsymbol{\theta}_V) \,\|\, p(\boldsymbol{f},\mathbf{u}|\mathbf{y})\right) &= \int p(\boldsymbol{f}|\mathbf{u})q(\mathbf{u}|\boldsymbol{\theta}_V)\log\frac{p(\boldsymbol{f},\mathbf{u}|\mathbf{y})}{p(\boldsymbol{f}|\mathbf{y})q(\mathbf{u}|\boldsymbol{\theta}_V)}d\mathbf{u}d\boldsymbol{f}\\
&= \int p(\boldsymbol{f}|\mathbf{u})q(\mathbf{u}|\boldsymbol{\theta}_V)\log\frac{\cancel{p(\boldsymbol{f}|\mathbf{u})}q(\mathbf{u}|\boldsymbol{\theta}_V)p(\mathbf{y})}{p(\mathbf{y}|\boldsymbol{f})\cancel{p(\boldsymbol{f}|\mathbf{u})}p(\mathbf{u})}\\
&= -\int p(\boldsymbol{f}|\mathbf{u})q(\mathbf{u}|\boldsymbol{\theta}_V)\log p(\mathbf{y}|\boldsymbol{f})d\boldsymbol{f}d\mathbf{u} + \log p(\mathbf{y})\\
&\quad + \mathrm{KL}\left(q(\mathbf{u}|\boldsymbol{\theta}_V) \,\|\, p(\mathbf{u})\right)\\
\log p(\mathbf{y}) &= \int q(\mathbf{u}|\boldsymbol{\theta}_V)\left[\int p(\boldsymbol{f}|\mathbf{u})\log p(\mathbf{y}|\boldsymbol{f})d\boldsymbol{f}\right]d\mathbf{u}\\
&\quad - \mathrm{KL}\left(q(\mathbf{u}|\boldsymbol{\theta}_V) \,\|\, p(\mathbf{u})\right) + \mathrm{KL}\left(p(\boldsymbol{f}|\mathbf{u})q(\mathbf{u}|\boldsymbol{\theta}_V) \,\|\, p(\boldsymbol{f},\mathbf{u}|\mathbf{y})\right)\\
&\geq \int q(\mathbf{u}|\boldsymbol{\theta}_V)\left[\int p(\boldsymbol{f}|\mathbf{u})\log p(\mathbf{y}|\boldsymbol{f})d\boldsymbol{f}\right]d\mathbf{u}\\
&\quad - \mathrm{KL}\left(q(\mathbf{u}|\boldsymbol{\theta}_V) \,\|\, p(\mathbf{u})\right) \triangleq L(q_{\mathbf{u}}).
\end{aligned}
$$

where $L(q_{\mathbf{u}})$ denotes a lower bound on the true log marginal likelihood, i.e. the ELBO. The expensive to compute $\mathrm{KL}\left(p(\boldsymbol{f}|\mathbf{u})q(\mathbf{u}|\boldsymbol{\theta}_V) \,\|\, p(\boldsymbol{f},\mathbf{u}|\mathbf{y})\right)$ is dropped to form this lower bound as it is strictly positive. Again $\log p(\mathbf{y})$ is constant with respect to all variational parameters of the model, and so by maximising $L(q_{\mathbf{u}})$ with respect to variational parameters $\boldsymbol{\theta}_V$, the divergence $\mathrm{KL}\left(p(\boldsymbol{f}|\mathbf{u})q(\mathbf{u}|\boldsymbol{\theta}_V) \parallel p(\boldsymbol{f}, \mathbf{u}|\mathbf{y})\right)$ is minimised; this brings the variational approximation to the posterior, closer to the true posterior.

If the likelihood $p(\mathbf{y}|\boldsymbol{f})$ factorises, the inner integral term becomes factorised across the data,

$$\log p(\mathbf{y}) \geq \int q(\mathbf{u}|\boldsymbol{\theta}_V) \left[\sum_{i=1}^{n} \int p(f_i|\mathbf{u}) \log p(y_i|f_i) df_i\right] d\mathbf{u} - \mathrm{KL}\left(q(\mathbf{u}|\boldsymbol{\theta}_V) \parallel p(\mathbf{u})\right) \quad (3.16)$$

From this bound the variational distribution $q(\mathbf{u}|\boldsymbol{\theta}_V)$ may be handled in one of two ways. The original approach to inference in the sparse variational Gaussian process model (Titsias, 2009), then differentiates this bound, with respect to $q(\mathbf{u}|\boldsymbol{\theta}_V)$ to find its optimal form. The optimal distribution for $q(\mathbf{u}|\boldsymbol{\theta}_V)$ turns out to be a Gaussian distribution,

$$q(\mathbf{u}|\boldsymbol{\theta}_V) = \mathcal{N}\left(\mathbf{u}|\mathbf{K}_{\mathbf{uu}}(\mathbf{K}_{\mathbf{uu}} + \beta\mathbf{K}_{\mathbf{u}\boldsymbol{f}}\mathbf{K}_{\boldsymbol{f}\mathbf{u}})^{-1}\mathbf{K}_{\boldsymbol{f}\mathbf{u}}\beta\mathbf{y}, \mathbf{K}_{\mathbf{uu}}(\mathbf{K}_{\mathbf{uu}} + \beta\mathbf{K}_{\mathbf{u}\boldsymbol{f}}\mathbf{K}_{\boldsymbol{f}\mathbf{u}})^{-1}\mathbf{K}_{\mathbf{uu}}\right). \quad (3.17)$$

Unfortunately by finding the optimal distribution for $q(\mathbf{u}|\boldsymbol{\theta}_V)$, and performing the integral analytically, the variational lower bound becomes coupled between data,

$$\log p(\mathbf{y}|\mathbf{X}) \geq \log \mathcal{N}\left(\mathbf{y}|\mathbf{0}, \beta^{-1}\mathbf{I} + \mathbf{K}_{\boldsymbol{f}\mathbf{u}}\mathbf{K}_{\mathbf{uu}}^{-1}\mathbf{K}_{\mathbf{u}\boldsymbol{f}}\right) - \frac{\beta}{2}\mathrm{tr}\left(\mathbf{K}_{\boldsymbol{f}\boldsymbol{f}} - \mathbf{K}_{\boldsymbol{f}\mathbf{u}}\mathbf{K}_{\mathbf{u}\boldsymbol{f}}\right).$$

This bound must be subsequently optimised with respect to the kernel hyper-parameters $\boldsymbol{\theta}_K$, and likelihood variance, β^{-1}. Since the bound now requires all the data to be considered at once, each evaluation of the lower bound, or its derivatives, is $\mathcal{O}(nm^2)$. Although this is already significantly less costly than the original $\mathcal{O}(n^3)$ complexity of Gaussian process regression, it still contains the term n that may be problematic on really large datasets. For a more thorough derivation of the above result, we refer the reader to the excellent review of Damianou (2015).

Hensman et al. (2013) noticed that, prior to "collapsing" the optimal distribution for $q(\mathbf{u}|\boldsymbol{\theta}_V)$ by finding its optimal form, Equation (3.17), and replacing it back into the lower bound, Equation (3.16), the bound factorises across data, and then so does its derivatives. In this case mini-batches of data may be used to obtain approximate gradients for the kernel hyper-parameters, likelihood parameters, and variational parameters of $q(\mathbf{u}|\boldsymbol{\theta}_V)$; this methodology is known as stochastic variational inference (SVI) (Hoffman et al., 2013). Alternatively the full lower bound can be evaluated by computing the a series of lower bound contributions, each on a mini-batch of which the union contains the whole data, and then summing the resulting contributions. The Gaussian process SVI model tries to learn variational parameters $\boldsymbol{\theta}_V$ directly, that capture the information within the variational posterior, $q(\mathbf{u}|\boldsymbol{\theta}_V)$, rather than using the optimal form that requires $\mathcal{O}(nm^2)$ computation to compute the lower bound and its derivatives.

With this approach, by selecting a mini-batch size of n_b, the resulting computational complexity becomes $\mathcal{O}(n_b m^2)$ for each step of the optimisation. A larger batch size will require more computation time, however more accurate estimates of the gradients will be obtained and so less optimisation steps should be needed. This approach is built upon in the following chapter, in which we consider the case where the likelihood $p(\mathbf{y}|\boldsymbol{f})$ is not Gaussian, and further may depend on more than a single latent function, $\boldsymbol{f}$; this will provide additional flexibility beyond existing survival models.

Prediction

The approximate predictive distribution for the sparse Gaussian process (Quinonero-Candela et al., 2007), is obtained by using the approximate posterior $q(\mathbf{u}|\boldsymbol{\theta}_V)$,

$$\begin{aligned}
p(\boldsymbol{f}^*|\mathbf{y}) &= \int p(\boldsymbol{f}^*|\boldsymbol{f})p(\boldsymbol{f}|\mathbf{u},\mathbf{y})p(\mathbf{u}|\mathbf{Z},\mathbf{y})d\boldsymbol{f}d\mathbf{u}\\
&\approx \int p(\boldsymbol{f}^*|\boldsymbol{f})p(\boldsymbol{f}|\mathbf{u})q(\mathbf{u}|\boldsymbol{\theta}_V)d\boldsymbol{f}d\mathbf{u}\\
&= \int \mathcal{N}\left(\boldsymbol{f}^*|\mathbf{K}_{\boldsymbol{f}^*\mathbf{u}}\mathbf{K}_{\mathbf{uu}}^{-1}\mathbf{u}, \mathbf{K}_{\boldsymbol{f}^*\boldsymbol{f}^*} - \mathbf{K}_{\boldsymbol{f}^*\mathbf{u}}\mathbf{K}_{\mathbf{uu}}^{-1}\mathbf{K}_{\mathbf{u}\boldsymbol{f}^*}\right) q(\mathbf{u}|\boldsymbol{\theta}_V)d\mathbf{u} &\quad (3.18)\\
&= \mathcal{N}\left(\boldsymbol{f}^*|\mathbf{K}_{\boldsymbol{f}^*\mathbf{u}}\mathbf{K}_{\mathbf{uu}}^{-1}\boldsymbol{\mu}_V, \mathbf{K}_{\boldsymbol{f}^*\boldsymbol{f}^*} - \mathbf{K}_{\boldsymbol{f}^*\mathbf{u}}(\mathbf{K}_{\mathbf{uu}}^{-1} - \mathbf{K}_{\mathbf{uu}}^{-1}\boldsymbol{\Sigma}_V\mathbf{K}_{\mathbf{uu}}^{-1})\mathbf{K}_{\mathbf{u}\boldsymbol{f}^*}\right) &\quad (3.19)
\end{aligned}$$

where $\boldsymbol{\mu}_V$ and $\boldsymbol{\Sigma}_V$ are the mean and covariance of the approximate posterior $q(\mathbf{u}|\boldsymbol{\theta}_V)$. This result will be used again in the following chapter.

By using the optimal form of $q(\mathbf{u}|\boldsymbol{\theta}_V)$, as in Equation (3.17), the predictive distribution of Equation (3.19) can be evaluated in closed form,

$$\begin{aligned}
p(\boldsymbol{f}^*|\mathbf{y}) &= \mathcal{N}\left(\boldsymbol{f}^*|\boldsymbol{\mu}^*, \boldsymbol{\Sigma}^*\right)\\
\boldsymbol{\mu}^* &= \mathbf{K}_{\boldsymbol{f}^*\mathbf{u}}(\mathbf{K}_{\mathbf{uu}} + \beta\mathbf{K}_{\mathbf{u}\boldsymbol{f}}\mathbf{K}_{\boldsymbol{f}\mathbf{u}})^{-1}\mathbf{K}_{\boldsymbol{f}\mathbf{u}}\beta\mathbf{y}\\
\boldsymbol{\Sigma}^* &= \mathbf{K}_{\boldsymbol{f}^*\boldsymbol{f}^*} - \mathbf{K}_{\boldsymbol{f}^*\mathbf{u}}(\mathbf{K}_{\mathbf{uu}}^{-1} - (\mathbf{K}_{\mathbf{uu}} + \beta\mathbf{K}_{\mathbf{u}\boldsymbol{f}}\mathbf{K}_{\boldsymbol{f}\mathbf{u}})^{-1})\mathbf{K}_{\mathbf{u}\boldsymbol{f}^*}
\end{aligned}$$

3.4 Conclusion

This chapter has reviewed the key components of the Gaussian process framework. Their non-parametric nature, and use as flexible priors over non-linear functions, are the main source of their applicability. They provide the ability to express a wide range of complex prior beliefs about unknown functions through the kernel function. They also trivially provide uncertainty about these latent functions, even when observed data is taken into account. Unfortunately in their standard form, they are both unable to handle non-Gaussian data, required for survival analysis, and scale poorly with respect to the number of training data they are exposed to. Approximations can however be made to partially overcome both of these difficulties as we have seen in this chapter. We will now consider how these models can be harnessed to overcome the limitations of survival analysis as set out in the preceding chapter.

4

Gaussian Process Survival Analysis

Several of the most popular approaches to survival analysis, Chapter 2, can be formulated from the perspective of generalised linear models, Section 2.3. An important restriction however is that these models typically model some parameter of the likelihood as a linear function of the covariates, or some transformation of covariates, $\mathbf{x}$, and infer the maximum likelihood solution for the latent function. In some cases this linear restriction is not correct for the particular data being studied. In the preceding chapter we saw that a popular approach within the machine learning community when posed with a problem that requires inference of an unknown function, is to first encode any prior information available within a Gaussian process prior. This prior can then be used to infer the corresponding posterior distribution, given the observed data $\mathbf{y}$. Gaussian process regression allows linear restrictions to be relaxed, and consequently provides a more powerful platform of inference than the standard generalised linear modelling framework. In this chapter we consider some existing work that has attempted to unify the areas of survival analysis with Gaussian processes inference. The first novel contribution is then given, an inference platform called *chained Gaussian processes*. The framework is more flexible in terms of the generative models for survival analysis it can handle, relaxing some of the restrictions of existing approaches to survival analysis with Gaussian processes. The provided inference method is also scalable, allowing large datasets to be handled. As with many inference methods in the machine learning literature, the framework is also applicable to a range of other applications. To make this clear a range of experiments are carried out on differing application areas alongside survival analysis. This shows the flexibility of this framework, beyond the scope of the survival analysis context for which it was developed.

4.1 Existing Gaussian Process Survival Models

There is some existing literature on applying the Gaussian process methodology to survival analysis (Barrett and Coolen, 2013; Joensuu et al., 2013, 2012; Li et al., 2016) though it is relatively limited. Very recently there has been an increase in attention within machine learning area on survival models, including deep learning approaches (Katzman et al., 2016; Ranganath et al., 2016), and a new Gaussian process method based on sampling and rejecting possible failure times (Fernandez et al., 2016). Its underlying model is strongly related to Joensuu et al. (2012) covered in Section 4.1.1. The new model uses a Gaussian process to model the effect of the covariates, though does not require a grid like approximation to the likelihood, and a parametric baseline function is used.

Although survival analysis has not been widely explored within the machine learning field. With a single event occurrence survival analysis can be seen as a special case of censored data, where the observation is also constrained to be positive. A number of authors have investigated the application of the Gaussian process framework in the context of censored data (Ertin, 2007; Groot and Lucas, 2012; Osborne, 2010; Osborne et al., 2010) that shares similar challenges to those used in survival analysis; namely the existence of a non-Gaussian likelihood, requiring approximations to the posterior of the latent function.

The collective survival of a group of individuals that can be pooled into subgroups, are related to work in the Poisson process and Log-Gaussian Cox process literature (Møller et al., 1998). The hazard function is analogical to the intensity function of Poisson processes; though in the case of the hazard function it is conditioned on the failure having not yet occurred. One of the later application areas of the chained Gaussian process, will be consider the case where the intensity function of a Log-Gaussian Cox process arises from multiple functions being combined. An example is provided in Section 4.4.4 whereby the homicide rate, i.e. the intensity of failures, within the city of Chicago is modelled as a sum of two positive intensity functions. In this case count data of homicides within discretised postcode areas are used to pool individual failures based on their location at the time of failure. This decomposition of intensities is a generative model that cannot be trivially handled by existing inference methods.

We will first introduce two existing likelihoods that have been used for survival analysis within the Gaussian process literature, discuss their limitations, and then in the subsequent sections we will propose a novel method that relaxes some of these limitations.

4.1.1 Piecewise Constant

In Section 2.4 the Cox proportional hazards (CPH) model was reviewed, and it was made clear that assuming a log linear relationship for the relative hazard $h_R(\boldsymbol{\theta}, \mathbf{x}_{i,:})$, is restrictive from a modelling perspective and it would be beneficial if it could be relaxed. The separation of the relative hazard function from the baseline hazard function, however has proven a popular approach. Royston and Lambert (2011) discusses approaches to extending beyond the log linear assumption of the relative hazard used in the CPH model, $h_R(\boldsymbol{\theta}, \mathbf{x}_{i,:}) = \exp(\boldsymbol{w}^\top \mathbf{x}_i)$, by fitting spline models for the non-linear function, $h_R(\boldsymbol{\theta}, \mathbf{x}_{i,:}) = \exp(g(\mathbf{x}_{i,:}, \boldsymbol{\theta}))$. However from a Bayesian perspective, spline models do not intrinsically handle the uncertainty associated with this function and confidence intervals must subsequently be computed using bootstrap methods.

Gaussian processes provide prior probability distributions over functions. The object of interest is then usually the Gaussian process posterior distribution of the latent function, given the data; such distributions intrinsically handle the uncertainty associated with these functions. It is natural then to question whether a Gaussian process prior could be placed over the latent baseline hazard and latent relative hazard function, of the CPH model.

The likelihood computation of a survival model posed in terms of the hazard function, $h_T(t|\boldsymbol{\theta}, \mathbf{x})$, Equation (2.11), requires the computation of the survival function, Section 2.1.3. The survival function in turn requires an integration over negative hazard function, Equation (2.6), with respect to time. Unfortunately it is not possible to integrate a non-linear function of a Gaussian process, e.g. $\exp(g(\mathbf{x}_{i,:}))$, analytically across an input range. If a Gaussian process prior is put over the baseline hazard rate, that is a function of time, this then creates an intractability in computing the likelihood.

Martino et al. (2010) proposed a Bayesian model where this problem is partially circumvented, by assuming a *piecewise log-constant baseline* (Breslow, 1972). The model assumes that within finite partitions of the time axis (interval), $0 = s_0 < s_1 < s_2 < \cdots < s_B$, with $s_B > t_i$ for all $i = 1, \cdots, n$ observed individuals failure times t_i, the baseline hazard is constant within each time interval s_i,

$$h_{0T}(t|\boldsymbol{\theta}) = \tau_b, \qquad \text{for } t \in [s_{b-1}, s_b] \qquad \text{for } b = 1, \cdots, B$$

Now within each interval, s_b, the hazard is constant, and so for $t_i \in (s_{b-1}, s_b)$,

$$h_T(t|\boldsymbol{\theta}, \mathbf{x}_{i,:}) = h_{0T}(t|\boldsymbol{\theta}) h_R(\boldsymbol{\theta}, \mathbf{x}) = \tau_b \exp(g(\mathbf{x}_{i,:})) = \exp(\log \tau_b + g(\mathbf{x}_{i,:})) = \exp(\eta_{ib})$$

where $\eta_{ib} \triangleq \log \tau_b + g(\mathbf{x}_{i,:})$. With a constant hazard in each interval the integral in the log-likelihood becomes a summation over segments. For an individual i, where $t_i \in (s_{b-1}, s_b)$, using Equation (2.12), the log likelihood contribution is given by

$$\begin{aligned}\log L &= \nu_i \log h_T(t|\boldsymbol{\theta}, \mathbf{x}_{i,:}) - \int_0^{t_i} h_T(k|\boldsymbol{\theta}, \mathbf{x}_{i,:})dk \\ &= \nu_i(\log \tau_b + g(\mathbf{x}_{i,:})) - (t_i - s_b)\tau_b \exp(g(\mathbf{x}_{i,:})) - \sum_{j=0}^{b-1}(s_{j-1} - s_j)\tau_j \exp(g(\mathbf{x}_{i,:})) \\ &= \nu_i\eta_{ib} - (t_i - s_b)\exp(\eta_{ib}) - \sum_{j=0}^{b-1}(s_{j-1} - s_j)\exp(\eta_{ij}),\end{aligned}$$

where similarly $\eta_{ij} \triangleq \log \tau_j + g(\mathbf{x}_{i,:})$.

This likelihood has formed a basis of several publications for applying the Gaussian processes framework to survival analysis problems. Martino et al. (2010) themselves assume the log of the baseline hazard follows a first order random walk (Rue and Held, 2005), equivalent to a Gaussian process with Brownian motion covariance function, and assign priors over the parameters of the linear function $g(\mathbf{x}_{i,:})$. Joensuu et al. (2012) began by extending this idea by assigning a Gaussian process prior to the relative hazard function $g(\mathbf{x}_{i,:})$. Joensuu et al. (2013) then further extended this approach such that the log of the baseline hazard and the relative hazard is a joint Gaussian process, with interactions between covariates and time.

By using a Gaussian process prior over the log relative hazard function, $g(\mathbf{x}_{i,:})$, implicit interactions between all covariates are taken into account. In a spline model (Royston and Lambert, 2011) these interactions would have to be explicitly represented. Additionally uncertainty in the functions is captured in the posterior distribution. These models have been successful in predicting probability of recurrence of gastrointestinal stromal tumours after surgery (Joensuu et al., 2012), and the predicted hazards have subsequently been used to optimise timings of CT scans during the follow up period of cancer patients (Joensuu et al., 2013). Since the likelihood used is non-Gaussian, approximations to the true posterior must be made, as covered in Section 3.2. Existing approaches have used the Laplace approximation, and as such scale poorly in the number of observations, n. The method of inference used in the existing literature then limits their applicability to larger datasets, that are likely to arise in the coming years. Although experiments have not been carried out, by augmenting the data it is possible to reframe this likelihood as a special case of a Poisson regression model, and so the novel approximation given in Section 4.2 could be applied providing a scalable alternative inference method for this survival likelihood choice.

4.1.2 Log-logistic Likelihood

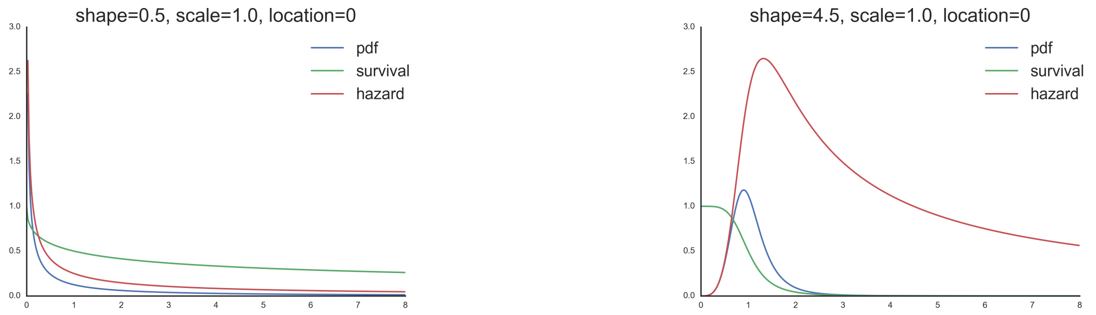
In Chapter 2 we saw that a popular model for survival analysis is the accelerated failure time class of models, Section 2.5, and can be considered within the class of generalised linear models. Specifying an appropriate noise model for AFT models is essential. A popular choice is the log-logistic distribution (Collett, 2015). This will be the likelihood that will serve as a basis for the survival analysis work carried out throughout the remainder of this work; though it should be noted all the proposed methods are equally applicable to other popular likelihoods such as the Weibull. In practice we find the log-logistic to be a flexible choice for a number of reasons explained below.

The log-logistic distribution has two parameters; a scale parameter, α, controlling the median of the distribution; and a shape parameter, β controlling the shape of the distribution. In existing AFT models, only the scale parameter is modelled as a function of the input. The scale parameter however must be restricted to be positive, and hence an exponential transformation function is usually chosen, $\alpha_i = \exp(g(\mathbf{x}_{i,:})) = \exp(\boldsymbol{w}^\top \mathbf{x}_{i,:})$. The corresponding likelihood function taking into account censored observations is given by,

$$\begin{aligned} p(\mathbf{y}|\boldsymbol{\alpha}, \beta, \boldsymbol{\nu}) &= \underbrace{\prod_{i=1}^{K:\nu=1} \frac{\left(\frac{\alpha_i}{\beta}\right)\left(\frac{y_i}{\alpha_i}\right)^{\beta-1}}{\left(1+\frac{y_i}{\alpha_i}^{\beta}\right)^2}}_{\text{failure time observed}} \underbrace{\prod_{j=1}^{M:\nu=0} \frac{1}{1+\left(\frac{y_j}{\alpha_j}\right)^{\beta}}}_{\text{censored individuals}} \\ &= \prod_{i=1}^{n} \left(\frac{\beta y_i^{\beta-1}}{\alpha_i}\right)^{\nu_i} \left(1+\left(\frac{y_i}{\alpha_i}\right)^{\beta}\right)^{-(\nu_i+1)} \end{aligned} \tag{4.1}$$

The log-logistic distribution has two convenient properties; probability density is only assigned to non-negative failure times, and the cumulative density function, required for computing the survival function, can be written in closed form.

By replacing the traditionally used (Collett, 2015) log linear function for the scale parameter with a latent function modelled as a log Gaussian process over the median failure, time can be affected in a flexible non-linear way by covariate values (Gelman et al., 2013); where as the standard GLM interpretation allows only a log linear effect. Similarly to the piecewise likelihood (Section 4.1.1) applied to Gaussian processes, approximations must also be used for the log-logistic likelihood Gaussian process



(a) Log-logistic, $\alpha = 1.0$, $\beta = 0.5$

(b) Log-logistic, $\alpha = 1.0$, $\beta = 4.5$

Fig. 4.1 Log-logistic distribution shape changing between exponential shaped hazard function, to unimodal shaped hazard function in response to changes in the shape parameter β

model to make progress. Existing approaches for the log-logistic AFT model have used the Laplace approximation, Section 3.2.1. Barrett and Coolen (2013) used a similar approach to this, where instead of assuming a log-logistic likelihood they used a likelihood that gives a AFT model very similar to a log-normal, with a slight modification of the log link function; they also used a Laplace approximation for inference of the intractable Gaussian process posterior.

There are two major restrictions of this model. Firstly the Laplace approximation does not scale well with the number of data, $\mathcal{O}(n^3)$ if no low-rank matrix approximations are used. Secondly only the scale parameter is assumed to be input-dependent.

The log-logistic distribution has the property that when $\beta > 1$ the failure density distribution is unimodal, when $0 < \beta \leq 1$ it is exponential, illustrated in Figure 4.1. Existing approaches have assumed this shape parameter to be the same for all individuals in the cohort. This equivalently means the hazard function of an individual can either decrease over time, or rise to a maximum and then decrease over time (Kleinbaum and Klein, 2006). However it does not allow the overall shape of the hazard function to change in response to the individuals covariate values.

In the remainder of this chapter, we propose an inference method that will overcome both of these limitations allowing a flexible survival analysis model to be used, known as the chained survival analysis model. Elements of this method have previously been published in Saul et al. (2016). The model allows all parameters of survival likelihood, namely the log-logistic likelihood, to able to respond to changes in the input, $\mathbf{x}$. This includes both the scale parameter, α, *and* the shape parameter, β. During the derivation, it will become clear this method is not purely applicable to

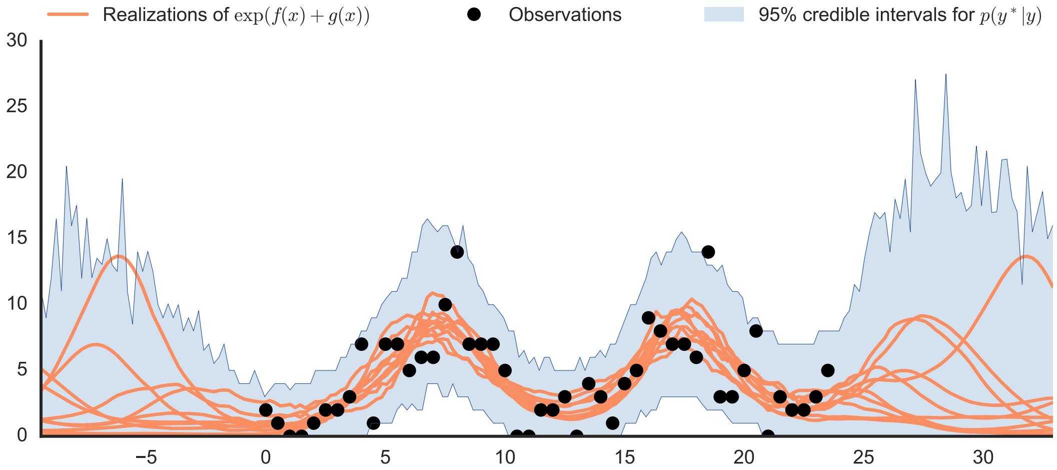


Fig. 4.2 Posterior distribution for Poisson distribution, where the exponential transformation function ensures positivity

survival analysis, and that it can indeed be used in conjunction with any likelihood that requires multiple parameters to be input-dependent functions.

4.2 Chained Gaussian Processes

Many examples of the flexibility of Gaussian process models as priors over non-linear functions have been shown, including in a survival analysis setting. However it has been made clear in this chapter that the assumption that observations are a *Gaussian* corruption of a *single function* is too restrictive in the context of survival analysis; it is necessary to allow non-Gaussian likelihoods to be used. The inability to specify multiple latent functions has also been shown to be restrictive from a generative modelling perspective; the shape of the failure time distribution, and corresponding hazard function, in the context of the log-logistic likelihood depends on two parameters that may change dependent on the individual. The dependence on multiple parameters is a property shared by other likelihoods used for survival analysis and beyond. Many appropriate likelihoods for survival analysis also require parameter restrictions, such as positivity. In a generalised linear model, Section 2.3, a link function is used to connect a parameter of interest in the likelihood, such as its mean or rate parameter, with the function modelling it. Typically the function that the link function acts on is restricted to be linear, but as in the preceding section, it can be modelled with a Gaussian process. It is also typically assumed in GLMs that only one parameter of the likelihood is input-dependent.

Writing models in terms of the link function, λ, captures the linear nature of the underlying model, but it is somewhat against the probabilistic approach to modeling where we consider the *generative model* of our data. While there's nothing wrong with this relationship mathematically, when we consider the generative model we never apply the link function directly, we consider the transformation function. For example, consider two input-dependent latent functions, $\boldsymbol{f} = f(\mathbf{x})$ and $\boldsymbol{g} = g(\mathbf{x})$. The log link can be used to connect the rate in a Poisson distribution with a GP, $\log \boldsymbol{\lambda} = \boldsymbol{f} + \boldsymbol{g}$. For the log link, the transformation function is the exponential, $\boldsymbol{\lambda} = \exp(\boldsymbol{f} + \boldsymbol{g})$. Writing the model in this form emphasizes the importance that the transformation function has on the generative model (see e.g. work on warped GPs (Snelson et al., 2004)). For the Poisson example, the exponential transformation function acts to ensure the mean of the Poisson distribution is positive; alternatively a sigmoid transformation function may be used, the positivity constraint is the important aspect. An example whereby $\boldsymbol{f} + \boldsymbol{g}$ is modelled with a Gaussian process is illustrated in Figure 4.2. The log link implies a multiplicative combination of $\boldsymbol{f}$ and $\boldsymbol{g}$, as follows $\boldsymbol{\lambda} = \exp(\boldsymbol{f} + \boldsymbol{g}) = \exp(\boldsymbol{f}) \exp(\boldsymbol{g})$. The Poisson distribution is used to model count data, for example the number of deaths within a given area. By using the product of two functions, this suggests that the intensity of deaths is a product of two separate positive functions. In some cases we may wish to consider an additive model, $\boldsymbol{\lambda} = \exp(\boldsymbol{f}) + \exp(\boldsymbol{g})$; for example if the intensity of deaths is the sum of two separate positive functions. Since this function is non-invertible there is no corresponding link function applied to $\boldsymbol{\lambda}$ that renders the two underlying processes additive. Such a model no longer falls within the class of generalised linear models. An example of how the model that follows can handle this generative modelling assumption in the context of Poisson regression for decomposable homicide intensity rates is given in Section 4.4.4.

Similarly, by modelling both the scale and the shape parameters of the log-logistic AFT likelihood as input-dependent, whilst maintaining positivity, there is no corresponding link function that allows a GLM interpretation, as in Equation 2.16. In this case it too cannot be considered inside the class of generalised linear models.

More generally we are interested in performing inference where there are multiple input-dependent functions modelling likelihood parameters, and the likelihood function is non-Gaussian. In doing so, the model should be capable of extending to other likelihoods also commonly used in survival analysis; not just the log-logistic distribution that will be to focus of the experiments.

The flexibility provided by the model that is to follow will additionally allow fewer assumptions to be made during modelling. The focus will be primarily on likelihoods,

$p(\mathbf{y}|\boldsymbol{f}, \boldsymbol{g})$, that contain two input-dependent latent parameters, $\boldsymbol{f}$ and $\boldsymbol{g}$ respectively, such as the log-logistic likelihood; though the model is general enough to handle more. Parameters of interest could be a latent mean, the latent median, or the shape parameter of the distribution, amongst others.

Almost all likelihoods require multiple parameters to be learnt. Traditionally in the Gaussian process literature, the model expects one such parameter to change with respect to the input space. For the other parameters MAP solutions are typically used, or alternatively these parameters are integrated out approximately (Rue et al., 2009). In this work we accept that other parameters may change as a function of the input space, and look at inferring posterior Gaussian process functions for these parameters as well. We will assign a Gaussian process prior for the two latent functions of interest, $\boldsymbol{f} \sim \mathcal{N}(\boldsymbol{\mu}_f, \mathbf{K}_{\boldsymbol{ff}})$, and $\boldsymbol{g} \sim \mathcal{N}(\boldsymbol{\mu}_g, \mathbf{K}_{\boldsymbol{gg}})$ respectively.

In the inference method that follows, issues arising from this extension beyond standard Gaussian process regression will be addressed by developing a novel variational approximation. This will provide a framework for *non-linear combination* of multiple latent processes; allowing the modelling restrictions of a single input-dependent likelihood parameter, and Gaussian likelihood to be overcome. Because these models cannot be written in the form a single link function we call this approach *chained Gaussian processes*, as it is an extension beyond the use of a single link function. The flexibility of the model will be applied to the survival analysis context, producing a *chained survival analysis* model. For non-Gaussian likelihoods including the log-logistic, restrictions on the latent function values may differ, and a non-linear transformation of the latent functions, $\boldsymbol{f} \in \mathbb{R}^{n \times p}$, $\boldsymbol{g} \in \mathbb{R}^{n \times p}$ are often required. The inference method will overcome the difficulties introduced by the non-normality of the likelihood and the nonexistence of a single link function for both latent functions in a scalable way with sparse variational methods, Section 3.3. It will also provide the capability of using stochastic gradients during inference. Scalability is desirable as parameters may only become well determined as the number of observations grow large.

To render the model tractable the inference procedure extends recent advances in large scale variational inference approaches to Gaussian processes (Hensman et al., 2013). The inference approach builds on work by Hensman et al. (2015b) for handling non-Gaussian likelihoods, that in turn builds on the variational inference method proposed by Opper and Archambeau (2009) reviewed in Section 3.2.2.

We will begin by deriving the appropriate variational bound and then carry out experiments for the chained survival analysis model, Section 4.3.1. The novel framework will then also be applied to a variety of other Gaussian likelihoods, showing that the method is flexible enough to handle both likelihoods the contain multiple input-dependent parameters and, as in the Poisson example above, one parameter that is to be modelled by some function of multiple input-dependent functions.

4.2.1 Variational Bound

Sparse approximations of Gaussian processes, Section 3.3, serve as a basis for the non-Gaussian likelihood model that follows. For non-Gaussian likelihoods, even with a single latent process, the marginal likelihood, $p(\mathbf{y})$, is not tractable; but it can be lower bounded variationally using a factorised posterior. Section 3.2 discussed one such variational approach. From here on we assume that the latent functions, $\boldsymbol{f} = f(\mathbf{x})$ and $\boldsymbol{g} = g(\mathbf{x})$ are *a priori* independent, and as in Section 3.3 are conditioned on a set of function specific inducing points, $\mathbf{u}_f$, $\mathbf{u}_g$. The prior independence assumption may be invalid, particularly if there should be a strong correlation between the parameters, though it is frequently the case that we do not know what type of prior dependence should be given, and so it is typically a reasonable assumption. The above assumptions give the following factorisation,

$$p(\boldsymbol{f}, \boldsymbol{g}|\mathbf{u}_f, \mathbf{u}_g) = p(\boldsymbol{f}|\mathbf{u}_f)p(\boldsymbol{g}|\mathbf{u}_g). \tag{4.2}$$

The derivation of the variational lower bound then follows a similar form as (Hensman et al., 2015b) with the extension to multiple latent functions. We begin by writing down our log marginal likelihood,

$$\log p(\mathbf{y}) = \log \int p(\mathbf{y}|\boldsymbol{f}, \boldsymbol{g})p(\boldsymbol{f}, \boldsymbol{g}|\mathbf{u}_f, \mathbf{u}_g)p(\mathbf{u}_f)p(\mathbf{u}_g)d\boldsymbol{f}\, d\boldsymbol{g}\, d\mathbf{u}_f\, d\mathbf{u}_g$$

then introduce a factorised variational approximation to the posterior of the form,

$$p(\boldsymbol{f}, \boldsymbol{g}, \mathbf{u}_g, \mathbf{u}_f|\mathbf{y}) \approx p(\boldsymbol{f}|\mathbf{u}_f)p(\boldsymbol{g}|\mathbf{u}_g)q(\mathbf{u}_f)q(\mathbf{u}_g), \tag{4.3}$$

where we have made the additional assumption that the latent functions also factorise in the variational posterior. This factorisation assumption may not always be strictly valid, but it is required in order to allow the inference to be tractable. If there should only be very weak correlation between the parameters the model should not be badly affected by this approximation. If the parameters however should be strongly correlated, the modelling assumptions would be poor and this could result in spurious inferences and convergence difficulties, so care must be taken. It may be the case that a re-parameterisation of the model can allow it to be written in terms of more independent parameters, allowing the factorisation assumption to be valid. In practice we find that inference can still be effective even when there would naturally be some correlation between parameters being modelled.

Using Jensen's inequality and the factorisation of the latent functions, Equation (4.2), a variational lower bound can then be obtained for the log marginal likelihood,

$$\begin{aligned}\log p(\mathbf{y}) &= \log \int p(\mathbf{y}|\boldsymbol{f}, \boldsymbol{g}) p(\boldsymbol{f}|\mathbf{u}_f) p(\boldsymbol{g}|\mathbf{u}_g) p(\mathbf{u}_f) p(\mathbf{u}_g) d\boldsymbol{f}\, d\boldsymbol{g}\, d\mathbf{u}_f\, d\mathbf{u}_g \\ &\geq \int q(\boldsymbol{f}) q(\boldsymbol{g}) \log p(\mathbf{y}|\boldsymbol{f}, \boldsymbol{g}) d\boldsymbol{f}\, d\boldsymbol{g} - \mathrm{KL}\left(q(\mathbf{u}_f) \,\|\, p(\mathbf{u}_f)\right) - \mathrm{KL}\left(q(\mathbf{u}_g) \,\|\, p(\mathbf{u}_g)\right),\end{aligned} \tag{4.4}$$

where $q(\boldsymbol{f}) = \int p(\boldsymbol{f}|\mathbf{u}_f) q(\mathbf{u}_f) d\mathbf{u}_f$ and $q(\boldsymbol{g}) = \int p(\boldsymbol{g}|\mathbf{u}_g) q(\mathbf{u}_g) d\mathbf{u}_g$. Assuming Gaussian process priors on the latent functions we recover the conditionals,

$$\begin{aligned}p(\boldsymbol{f}|\mathbf{u}_f) &= \mathcal{N}\left(\boldsymbol{f}|\mathbf{K}_{\boldsymbol{f}\mathbf{u}_f}\mathbf{K}_{\mathbf{u}_f\mathbf{u}_f}^{-1}\mathbf{u}_f, \mathbf{K}_{\boldsymbol{f}\boldsymbol{f}} - \mathbf{Q}_{\mathbf{ff}}\right) \\ p(\boldsymbol{g}|\mathbf{u}_g) &= \mathcal{N}\left(\boldsymbol{g}|\mathbf{K}_{\boldsymbol{g}\mathbf{u}_g}\mathbf{K}_{\mathbf{u}_g\mathbf{u}_g}^{-1}\mathbf{u}_g, \mathbf{K}_{\boldsymbol{g}\boldsymbol{g}} - \mathbf{Q}_{\boldsymbol{g}\boldsymbol{g}}\right),\end{aligned}$$

where

$$\begin{aligned}\mathbf{Q}_{\mathbf{ff}} &= \mathbf{K}_{\boldsymbol{f}\mathbf{u}_f}\mathbf{K}_{\mathbf{u}_f\mathbf{u}_f}^{-1}\mathbf{K}_{\mathbf{u}_f\boldsymbol{f}} \\ \mathbf{Q}_{\boldsymbol{g}\boldsymbol{g}} &= \mathbf{K}_{\boldsymbol{g}\mathbf{u}_g}\mathbf{K}_{\mathbf{u}_g\mathbf{u}_g}^{-1}\mathbf{K}_{\mathbf{u}_g\boldsymbol{g}}.\end{aligned}$$

Note that kernel functions for $\boldsymbol{f}$ and $\boldsymbol{g}$, can differ though their inducing input locations, $\mathbf{Z}$, are shared.

We take $q(\mathbf{u}_f)$ and $q(\mathbf{u}_g)$ to be Gaussian distributions with variational parameters, $q(\mathbf{u}_f) = \mathcal{N}\left(\mathbf{u}_f|\boldsymbol{\mu}_{fu}, \boldsymbol{S}_f\right)$ and $q(\mathbf{u}_g) = \mathcal{N}\left(\mathbf{u}_g|\boldsymbol{\mu}_{gu}, \boldsymbol{S}_g\right)$. Using the properties of multivariate Gaussian distributions this results in tractable integrals for $q(\boldsymbol{f})$ and $q(\boldsymbol{g})$,

$$q(\boldsymbol{f}) = \mathcal{N}\left(\boldsymbol{f}|\mathbf{K}_{\boldsymbol{f}\mathbf{u}_f}\mathbf{K}_{\mathbf{u}_f\mathbf{u}_f}^{-1}\boldsymbol{\mu}_{fu}, \mathbf{K}_{\boldsymbol{f}\boldsymbol{f}} + \hat{\mathbf{Q}}_{\boldsymbol{f}\boldsymbol{f}}\right) \tag{4.5}$$

$$q(\boldsymbol{g}) = \mathcal{N}\left(\boldsymbol{g}|\mathbf{K}_{\boldsymbol{g}\mathbf{u}_g}\mathbf{K}_{\mathbf{u}_g\mathbf{u}_g}^{-1}\boldsymbol{\mu}_{gu}, \mathbf{K}_{\boldsymbol{g}\boldsymbol{g}} + \hat{\mathbf{Q}}_{\boldsymbol{g}\boldsymbol{g}}\right), \tag{4.6}$$

where

$$\begin{aligned}\hat{\mathbf{Q}}_{\boldsymbol{f}\boldsymbol{f}} &= \mathbf{K}_{\boldsymbol{f}\mathbf{u}_f}\mathbf{K}_{\mathbf{u}_f\mathbf{u}_f}^{-1}(\boldsymbol{S}_f - \mathbf{K}_{\mathbf{u}_f\mathbf{u}_f})\mathbf{K}_{\mathbf{u}_f\mathbf{u}_f}^{-1}\mathbf{K}_{\mathbf{u}_f\boldsymbol{f}} \\ \hat{\mathbf{Q}}_{\boldsymbol{g}\boldsymbol{g}} &= \mathbf{K}_{\boldsymbol{g}\mathbf{u}_g}\mathbf{K}_{\mathbf{u}_g\mathbf{u}_g}^{-1}(\boldsymbol{S}_g - \mathbf{K}_{\mathbf{u}_g\mathbf{u}_g})\mathbf{K}_{\mathbf{u}_g\mathbf{u}_g}^{-1}\mathbf{K}_{\mathbf{u}_g\boldsymbol{g}}.\end{aligned}$$

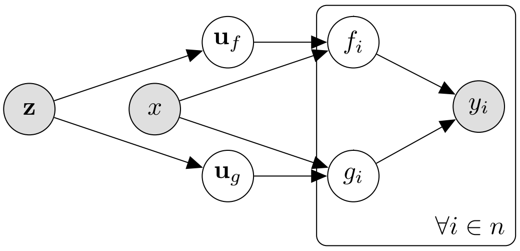


Fig. 4.3 Graphical model describing the chained GP model. In the posterior note $\boldsymbol{f}$ and $\boldsymbol{g}$ are integrated out.

The variational parameters, $\boldsymbol{\mu}_{fu}, \boldsymbol{\mu}_{gu}, \boldsymbol{S}_f, \boldsymbol{S}_g$ must be learnt through optimisation.

The KL terms in Equation (4.4) and their derivatives can be computed in closed form and are inexpensive as they are divergences between Gaussian distributions with $m \times m$ covariance matrices. However, an intractable integral, $\int q(\boldsymbol{f})q(\boldsymbol{g}) \log p(\mathbf{y}|\boldsymbol{f}, \boldsymbol{g}) d\boldsymbol{f}\, d\boldsymbol{g}$, still remains. Since we consider only likelihoods that completely factorise across data points, such as the log-logistic,

$$p(\mathbf{y}|\boldsymbol{f}, \boldsymbol{g}) = \textstyle\prod_{i=1}^{n} p(y_i|f_i, g_i),$$

the problematic integral in Equation (4.4) also factorises across data points, allowing us to use stochastic variational inference (Hensman et al., 2013; Hoffman et al., 2013),

$$\begin{aligned}\int q(\boldsymbol{f})q(\boldsymbol{g}) \log p(\mathbf{y}|\boldsymbol{f}, \boldsymbol{g}) d\boldsymbol{f}\, d\boldsymbol{g} &= \int q(\boldsymbol{f})q(\boldsymbol{g}) \log \prod_{i=1}^{n} p(y_i|f_i, g_i) d\boldsymbol{f}\, d\boldsymbol{g} \\ &= \textstyle\sum_{i=1}^{n} \displaystyle\int q(f_i)q(g_i) \log p(y_i|f_i, g_i) df_i\, dg_i. \end{aligned} \tag{4.7}$$

Note how this has reduced the computational burden from performing $b = 2$ infinite dimensional integrals, to a series of n, b dimensional Gaussian integrals over the log-likelihood of the data. In this case we are considering only two latent functions, $\boldsymbol{f}$ and $\boldsymbol{g}$ respectively, and so $b = 2$ however this can be trivially extended to more latent functions if necessary; though doing so requires increasingly high dimensional integrals. The final form of the variational lower bound then becomes,

$$\begin{aligned}\log p(\mathbf{y}) \geq \textstyle\sum_{i=1}^{n} \displaystyle\int q(f_i)q(g_i) \log p(y_i|f_i, g_i) df_i\, dg_i \\ - \mathrm{KL}\left(q(\mathbf{u}_f) \,\|\, p(\mathbf{u}_f)\right) - \mathrm{KL}\left(q(\mathbf{u}_g) \,\|\, p(\mathbf{u}_g)\right). \end{aligned} \tag{4.8}$$

Figure 4.3 shows a graphical model of the model prior to integration of $\boldsymbol{f}$ and $\boldsymbol{g}$.

The bound decomposes into a sum over data, as such the n input points can be visited in mini-batches, and the gradients and log-likelihood of each mini-batch can be subsequently summed. This operation of computing and summing can be also be parallelised (Dai et al., 2014; Gal et al., 2014). A single mini-batch can instead be visited obtaining a stochastic gradient for use in a stochastic optimisation (Hensman et al., 2013; Hoffman et al., 2013). This provides the ability to scale to huge datasets.

If the integral in Equation (4.7) and its gradients can be computed in an unbiased way, any factorising likelihood can be used. This is a key to its flexibility and its applicability to complex non-Gaussian likelihoods such as the log-logistic likelihood. In this case both the shape, β, and scale, α, parameters can now be modelled by input-dependent functions,

$$y_i \sim LogLogistic(\alpha_i = e^{f(\mathbf{x}_{i,:})}, \beta_i = e^{g(\mathbf{x}_{i,:})}),$$

where $f(\mathbf{x}) = \mathrm{GP}\left(\boldsymbol{\mu}_f, k_f(\mathbf{x}, \mathbf{x}')\right)$ and $g(\mathbf{x}) = \mathrm{GP}\left(\boldsymbol{\mu}_g, k_g(\mathbf{x}, \mathbf{x}')\right)$.

This can be seen as a chained Gaussian process since there is no single link function that allows the specification of this model under the modelling assumptions of a generalised linear model.

Using the variational bound above, all that is required is to perform a series of n two dimensional integrals, for both the log-likelihood and its gradients. Though these integrals are not analytic they can be approximated well with quadrature or Monte Carlo methods whilst maintaining a computationally feasible inference method when considering modest batch sizes.

A major strength of this method is that performing this integral is the only requirement to implement a model with a different likelihood, similarly to Hensman et al. (2015b); Nguyen and Bonilla (2014). Furthermore, since a stochastic optimiser is used, the gradients do not need to be exact. Our implementation can use off the shelf stochastic optimiser, such as Adagrad (Duchi et al., 2011) or RMSProp (Tieleman and Hinton, 2012). For many other likelihoods some portion of these integrals is analytically tractable; this will reduce the variance introduced by numerical integration required to evaluate the intractable integrals in Equation (4.7).

4.2.2 Quadrature and Monte Carlo

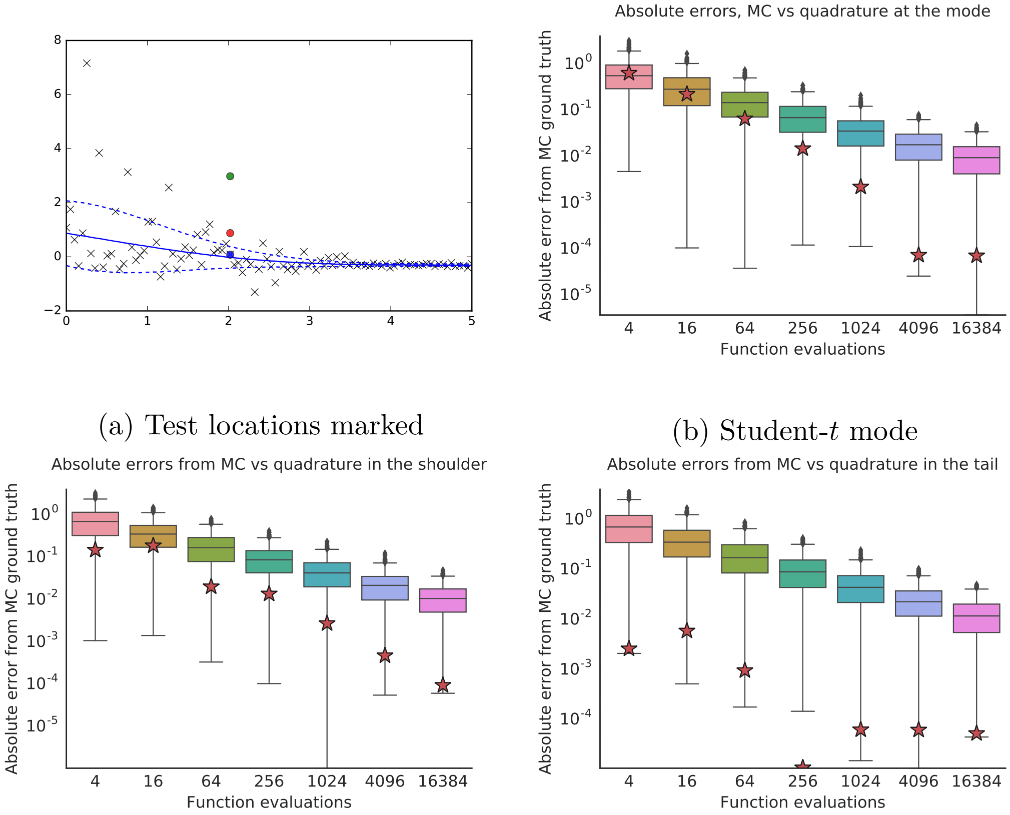
Computing Equation (4.7) requires a series of low-dimensional integrals. Recently, there has been progress in using stochastic methods to obtain unbiased estimates for such problems using centred representations (Kingma and Welling, 2014; Rezende et al., 2014). In this section, we investigate the effectiveness of Gauss-Hermite quadrature as a contrasting approach for two-dimensional integrals.

Gauss-Hermite quadrature approximates Gaussian integrals in one dimension using a pre-defined grid. For expectations of polynomial functions, the method is exact when the grid size meets the degree of the polynomial; for non-polynomial functions as we will encounter in general, we must accept a small amount of bias. To integrate higher dimensional functions, we must nest the quadrature, doing an integral across one dimension for each quadrature point in the other dimensions. Our experiments provide encouraging results in this case, suggesting the amount of bias is negligible. Figure 4.4 illustrates this, examining the accuracy of nested quadrature as compared to Monte Carlo (MC) estimates using the centered parameterisation (Kingma and Welling, 2014). Inspired by an examination of quadrature for expectation propagation (Jylänki et al., 2011), we examine the effectiveness for a several positions of the integral of a Student-t, a distribution that is problematic as it contains heavy tails.

Gauss-Hermite quadrature is appropriate for the integral in Equation (4.7) as the Gaussian posteriors $q(f_i)q(g_i)$ are convolved with a function $\log p(y_i|g_i, f_i)$. Monte Carlo integration is exact in the limit of infinite samples, however in practice a subset of samples must be used. Gauss-Hermite requires ph^b evaluations per point in the mini-batch, where h is the number of Gauss-Hermite points used, p is the number of output dimensions, and b is the number of latent functions. Since Monte Carlo is unbiased, using a stochastic optimiser with the stochastic estimates of the integral and its gradients will work effectively (Kingma and Welling, 2014; Nguyen and Bonilla, 2014), though we find the bias introduced by the quadrature approach to be negligible. For higher number of latent functions it may be more efficient to make use of low variance Monte Carlo estimates for the integrals. Gradients for the model can be computed in a similar way with the Gaussian identities used by Opper and Archambeau (2009), further details on gradients can be found in Appendix C.1.

4.2.3 Posterior and Predictive Distributions

From the results of the variational approximation, Section 3.2, and the posterior distribution assumptions made in Equation (4.3), when the variational lower bound has been maximised with respect to the variational parameters, $p(\mathbf{u}_f|\mathbf{y}) \approx q(\mathbf{u}_f)$ and



(a) Test locations marked

(b) Student-t mode

(c) Student-t shoulder

(d) Student-t tail

Fig. 4.4 Two dimensional Gauss-Hermite quadrature vs Monte Carlo. Each plot shows the log absolute error in estimating the two dimension integral required by our heteroscedastic Student-t model (see section 4.4.2), a likelihood with heavy tails and so particularly problematic for quadrature. In each case, the bias introduced by quadrature (circles) is small: a long way into the tail of the variance from the MC approximation. In fact, for small numbers of quadrature points, we often do better than the expected value using many more MC samples. Box-plots shows the absolute error on 1000 separate reruns of MC, whereas quadrature is deterministic. The error was evaluated at various points in the tail of the distribution as shown in a).

$p(\mathbf{u}_g|\mathbf{y}) \approx q(\mathbf{u}_g)$. The posterior for $p(\boldsymbol{f}^*|\mathbf{y}^*)$ under this approximation is

$$\begin{aligned} p(\boldsymbol{f}^*|\mathbf{y}) &= \int p(\boldsymbol{f}^*|\boldsymbol{f})p(\boldsymbol{f}|\mathbf{u}_f,\mathbf{y})p(\mathbf{u}_f|\mathbf{y})d\boldsymbol{f}\,d\mathbf{u}_f \\ &\approx \int p(\boldsymbol{f}^*|\mathbf{u}_f)q(\mathbf{u}_f)d\mathbf{u}_f = q(\boldsymbol{f}^*), \end{aligned}$$

and similarly so for $p(\boldsymbol{g}^*|\mathbf{y})$, where $q(\boldsymbol{f}^*)$ and $q(\boldsymbol{g}^*)$ become similar to Equation (4.5),

$$\begin{aligned} q(\boldsymbol{f}^*) &= \mathcal{N}\left(\boldsymbol{f}^*|\mathbf{K}_{\boldsymbol{f}^*\mathbf{u}_f}\mathbf{K}_{\mathbf{u}_f\mathbf{u}_f}^{-1}\boldsymbol{\mu}_f, \mathbf{K}_{\boldsymbol{f}^*\boldsymbol{f}^*} + \mathbf{K}_{\boldsymbol{f}^*\mathbf{u}_f}\mathbf{K}_{\mathbf{u}_f\mathbf{u}_f}^{-1}(\boldsymbol{S}_f - \mathbf{K}_{\boldsymbol{f}\boldsymbol{f}})\mathbf{K}_{\mathbf{u}_f\mathbf{u}_f}^{-1}\mathbf{K}_{\mathbf{u}_f\boldsymbol{f}^*}\right) \\ q(\boldsymbol{g}^*) &= \mathcal{N}\left(\boldsymbol{g}^*|\mathbf{K}_{\boldsymbol{g}^*\mathbf{u}_g}\mathbf{K}_{\mathbf{u}_f\mathbf{u}_f}^{-1}\boldsymbol{\mu}_f, \mathbf{K}_{\boldsymbol{g}^*\boldsymbol{g}^*} + \mathbf{K}_{\boldsymbol{g}^*\mathbf{u}_g}\mathbf{K}_{\mathbf{u}_f\mathbf{u}_f}^{-1}(\boldsymbol{S}_f - \mathbf{K}_{\boldsymbol{f}\boldsymbol{f}})\mathbf{K}_{\mathbf{u}_f\mathbf{u}_f}^{-1}\mathbf{K}_{\mathbf{u}_g\boldsymbol{g}^*}\right) \end{aligned}$$

Finally, treating each prediction point independently, the predictive distribution for each data pair $\{(\mathbf{x}_i^*, y_i^*)\}_{i=1}^{n^*}$ follows as

$$p(y_i^*|y_i,\mathbf{x}_i) = \int p(y_i^*|f_i^*, g_i^*)q(f_i^*)q(g_i^*)df_i^*\,dg_i^*.$$

This integral is analytically intractable in the general case, but again can be computed using a series of two dimensional quadrature or simple Monte Carlo sampling.

4.3 Chained Survival Analysis

A central goal of the development of the chained Gaussian process framework is to provide additional flexibility to the hazard function (Section 2.1.4) beyond existing models (Sections 4.1.2 and 4.1.1). The log-logistic distribution (Section 4.1.2) as discussed can be used as a likelihood for survival analysis models.

The log-logistic AFT based model given in Section 4.1.2, is however restrictive, as the *shape* of failure time distribution, β, is assumed to be the same for all individuals in the cohort. As noted in Section 4.1.2 when $\beta > 1$ the failure density distribution is unimodal, when $0 < \beta \leq 1$ it is exponential. By varying the shape parameter of the log-logistic distribution, the failure time distribution may vary between both a skewed unimodal and exponential shaped distributions. The corresponding hazard functions, that are a function of the failure time distribution, Equation (2.3b), will also then vary between a unimodal shaped hazard function, and an exponential shaped hazard function.

With the development of a chained survival model, the assumption that all patients share the same shape of failure time distribution can be relaxed by allowing the shape parameter, β of the log-logistic distribution to vary in response to the individuals covariates, as well as the scale parameter, α,

$$y_i \sim LL(\alpha_i = e^{f(\mathbf{x}_{i,:})}, \beta_i = e^{g(\mathbf{x}_{i,:})}),$$

where $f(\mathbf{x}) = \mathrm{GP}\left(\boldsymbol{\mu}_f, k_f(\mathbf{x}, \mathbf{x}')\right)$ and $g(\mathbf{x}) = \mathrm{GP}\left(\boldsymbol{\mu}_g, k_g(\mathbf{x}, \mathbf{x}')\right)$. Censoring can be taken into account similarly to Equation (4.1), resulting in a likelihood function of the form,

$$p(\mathbf{y}|\boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\nu}) = \underbrace{\prod_{i=1}^{K:\nu=1} \frac{\left(\frac{\alpha_i}{\beta_i}\right)\left(\frac{y_i}{\alpha_i}\right)^{\beta_i - 1}}{\left(1 + \frac{y_i}{\alpha_i}^{\beta_i}\right)^2}}_{\text{failure time observed}} \underbrace{\prod_{j=1}^{M:\nu=0} \frac{1}{1 + \left(\frac{y_j}{\alpha_j}\right)^{\beta_j}}}_{\text{censored individuals}} \tag{4.9}$$

where both $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ are now able to vary in response to the covariates. As in the chained Gaussian process methodology, we assume these parameters are independent, assign each a Gaussian process prior, and infer a posterior distribution of functions that represent how the parameters change in response to the input. Again there is no associated link-function in this case, and so the chained survival model fits inside the class of chained Gaussian process models.

4.3.1 Chained Survival Analysis Experiments

To evaluate the effectiveness of our chained survival analysis model, we applied the model to both a real dataset and synthetic dataset. The performance measure used is the negative log predictive density (NLPD) on held out data and 5-fold cross-validation is used. Since there are multiple latent functions that contain uncertainty, the NLPD is computed by sampling functions for both parameters repeatably; computing the NLPD for each sample, and averaging the NLPD over the samples. The non-linear optimisation of (hyper-) parameters is subject to local minima, as such multiple runs were performed on each fold with a range of parameter initialisations. The solution that obtained the highest log-likelihood on the training data of each fold was retained for subsequent prediction. The automatic relevance determination RBF kernel, Section 3.2, is used allowing one lengthscale per input dimension, in addition to a bias kernel[1]. Experiments are made with 100 inducing inputs, $\mathbf{Z}$, and their locations were optimised with respect to the lower bound of the log marginal likelihood following Titsias (2009).

The leukemia dataset (Henderson et al., 2002) contains censored failure times for 1043 leukemia patients and is known to have non-linear responses for certain covariates (Gelman et al., 2013).

[1] Code is publicly available at: https://github.com/SheffieldML/ChainedGP

Data	**NLPD**			
	G	**LSurv**	**VSurv**	**CHSurv**
leuk	$4.03 \pm .08$	$1.57 \pm .01$	$1.57 \pm .01$	$\mathbf{1.56 \pm .01}$
SimSurv	$5.45 \pm .06$	$2.52 \pm .02$	$2.52 \pm .02$	$\mathbf{2.16 \pm .02}$

Table 4.1 Results NLPD over 5 cross-validation folds with 10 replicates each, where ± represents the standard error of the mean. Models shown in comparison are sparse Gaussian (G), survival Laplace approximation (LSurv), survival variational approximation (VSurv), chained survival analysis model (CHSurv).

To generate the synthetic survival dataset, SimSurv, we first define latent functions that we wish to infer. These are a complex functions of a two dimensional set of covariates, $\mathbf{X} \in \mathbb{R}^{n\times 2}$,

$$\begin{aligned}\boldsymbol{\alpha} &= \exp\left(2\exp(-30(\mathbf{x}_{:,0} - \frac{1}{4})^2) + \sin(\pi \mathbf{x}_{:,1}^2) - 2\right)\\ \boldsymbol{\beta} &= \exp\left(\sin(2\pi\mathbf{x}_{:,0}) + \cos(2\pi\mathbf{x}_{:,1})\right).\end{aligned}$$

$n = 1000$ synthetic individuals are created, with covariates sampled uniformly from $\mathbf{x}_{i,0} \sim \text{Uniform}(0,1)$ and $\mathbf{x}_{i,1} \sim \text{Uniform}(0,1)$. True failure times, s_i, for each of the n individuals are then sampled from a log-logistic distribution as in Equation (4.9) using the corresponding parameter values for the individuals covariates as in Equations (4.4.1). Random dropout of the trial is simulated by censoring 20% of the failure times at a time uniformly drawn, $t_i \in [0, s_i]$, and the observed time is subsequently truncated to this time, $y_i = t_i$; a failure indicator $\nu_i = 0$ is also associated with the individual to indicate that the failure time was right censored. Otherwise the true time is used, $y_i = s_i$, and $\nu_i = 1$.

Figure 4.5 illustrates the flexibility of the chained survival model on one of the synthetic datasets generated. It is clearly able to capture the non-linearity of the shape and scale parameters over the two dimensional covariate information; which in turn leads to variation of the failure time distributions, and corresponding hazard functions, in response to the input covariates.

Table 4.1 shows the models performance on both a real and synthetic dataset. The model is compared to a standard sparse Gaussian process with no knowledge of the non-normality of the likelihood, a Gaussian process that uses a log-logistic likelihood with a Laplace approximation, as in Section 4.1.2, and a variational approximation matching that of Hensman et al. (2015b). Both the Laplace approximation and variational approximation are only able to find MAP estimates for the shape parameter; the

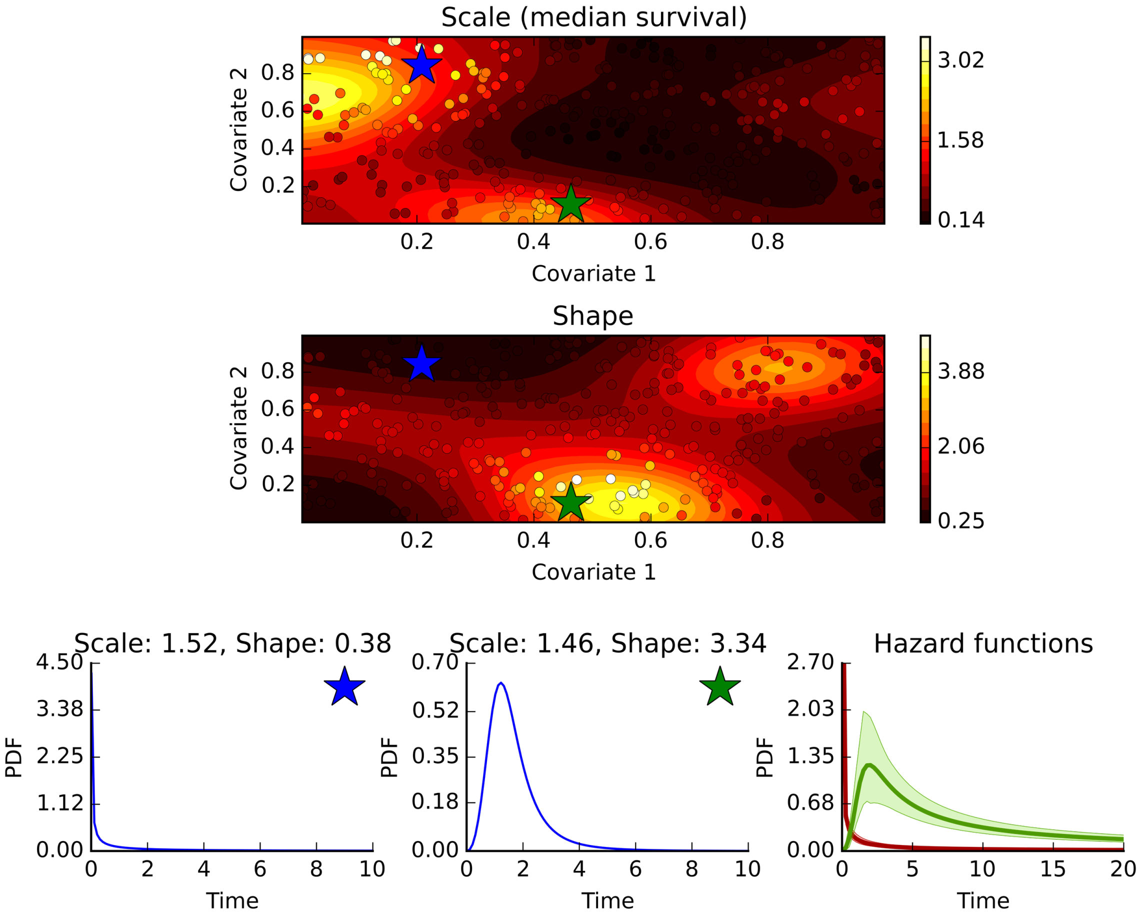


Fig. 4.5 Resulting model on synthetic survival dataset. Shows variation of median survival time and shape of log-logistic distribution, in response to differing covariate information. Background colour shows the chained-survivals predictions, coloured dots show ground truth. Lower figures show associated failure time distributions and hazards for two different synthetic patients. Depending on the input, the predicted shape of the hazard function can be either unimodal or exponential.

chained survival analysis model is able to allow the shape parameter to vary smoothly in response to the input.

On the leukemia dataset we find little advantage from using the chained survival model, but also note the model is robust on this example such that performance isn't degraded in this case, which is a very much desired property. This suggests that the additional flexibility does not impede the model. The lack of improvement indicates that, in this dataset, it is more than likely all individuals share the general shape of hazard function, and small changes attributed to different covariates cannot be inferred. It may also require additional data to infer these non-linearities, unlike existing Gaussian process based approaches, the chained survival model is capable of handling very large datasets, since the approach is highly scalable through the use of stochastic variational inference; though this is not demonstrated on this particular dataset due to its size. In the case of the synthetic dataset example however, a significant increase in performance is seen. Though only marginal improvements are found in this study by allowing $\boldsymbol{\beta}$ to be non-constant, there are a number of situations where we expect the shape of the distribution to change in response to the input. For example, if a study contained a cohort in which different patients were at different stages of infection, characterised by a covariate. If a patient had already started showing symptoms, it may be that their failure time distribution is exponential shaped, and they are at highest risk at the beginning of the study. If the oncome of symptoms have not yet started, the patient is may be currently at low risk, but this risk will rise at some point in the future, requiring a unimodal distribution.

It should be noted that since the number of training data used in these experiments is relatively small, it is not necessarily to use a sparse approximation; an approach more similar to the work of Opper and Archambeau (2009) could be instead modified do handle non-Gaussian likelihoods and multiple latent functions in a similar manner to this work. We hope in future work to demonstrate the applicability of the scalable model on larger survival datasets, though as shown in the following section, the model is able to scale computationally.

4.4 Further Experimental Results

Although the focus of the chained Gaussian process model is relax limitations of existing survival analysis methods, namely by allowing both the shape and scale of the failure density time distribution and hazard time to vary non-linearly in response to observed covariates; the model is also applicable to a wider range of likelihoods than those commonly used survival analysis.

We now provide some results of applying the inference procedure to a number of models with other likelihoods that contain multiple latent parameters. Some of these models also relate directly to modelling of failure data, such as Poisson distributed failure counts within a given area, others can be related to a variety of tasks within clinical data analysis. We also consider the Gaussian likelihood which makes clear the relationship between the chained Gaussian process models and existing work in heteroscedastic Gaussian process regression.

The experimental setup is similar to that of the chained survival analysis experiments, evaluating the performance of the model on both real and synthetic datasets, measured by NLPD on held out data table 4.2.

4.4.1 Heteroscedastic Gaussian

One alternative application of this framework is to the heteroscedastic Gaussian process model. Assume a Gaussian process prior for $\boldsymbol{f} \sim \mathcal{N}\left(\boldsymbol{\mu}_f, \mathbf{K}_{\boldsymbol{ff}}\right)$. If the likelihood distribution is a i.i.d Gaussian, $p(\mathbf{y}|\boldsymbol{f}, \boldsymbol{g}) = \prod_{i=1}^{n} \mathcal{N}\left(y_i|f_i, g_i\right)$, then we have two special cases. When g_i is independent of $\mathbf{X}$, e.g. $g_i = \beta^{-1}$ we have a Gaussian process with the conjugate homogeneous Gaussian likelihood, Section 3.1. When g_i changes as a function of $\mathbf{X}$, e.g. $g_i = e^{g(\mathbf{x})}$ we obtain a model for a heteroscedastic Gaussian process (Goldberg et al., 1997). It is common (Kersting et al., 2007; Lazaro-Gredilla and Titsias, 2011) to assume a log Gaussian process prior for the noise variance parameter, $\boldsymbol{g} \sim \mathcal{N}\left(\boldsymbol{\mu}_g, \mathbf{K}_{\boldsymbol{gg}}\right)$ to maintain positivity.

In the case that the likelihood is Gaussian, the integral in Equation 4.7 becomes analytic (Lazaro-Gredilla and Titsias, 2011),

$$\begin{aligned}
&\int q(f_i)q(g_i)\log p(y_i|f_i, g_i)df_i\, dg_i \\
&\quad = \int \mathcal{N}\left(f_i|m_{f_i}, v_{f_i}\right)\mathcal{N}\left(g_i|m_{g_i}, v_{g_i}\right)\log \mathcal{N}\left(y_i|f_i, e^{g_i}\right) \\
&\quad = \log \mathcal{N}\left(y_i|m_{f_i}, e^{m_{g_i} - \frac{v_{g_i}}{2}}\right) - \frac{v_{g_i}}{4} - \frac{v_{f_i}e^{-m_{g_i} + \frac{v_{g_i}}{2}}}{2}
\end{aligned}$$

where we define

$$\begin{aligned}
\boldsymbol{m}_f &= \mathbf{K}_{\boldsymbol{f}\mathbf{u}_f}\mathbf{K}_{\mathbf{u}_f\mathbf{u}_f}^{-1}\boldsymbol{\mu}_f & \boldsymbol{V}_f &= \mathbf{K}_{\boldsymbol{ff}} + \hat{\mathbf{Q}}_{\boldsymbol{ff}} \\
\boldsymbol{m}_g &= \mathbf{K}_{\boldsymbol{g}\mathbf{u}_g}\mathbf{K}_{\mathbf{u}_g\mathbf{u}_g}^{-1}\boldsymbol{\mu}_g & \boldsymbol{V}_g &= \mathbf{K}_{\boldsymbol{gg}} + \hat{\mathbf{Q}}_{\boldsymbol{gg}}.
\end{aligned}$$

Data	**NLPD**				
	G	**CHG**	**Lt**	**Vt**	**CHt**
Elevators1000	0.23 ± 0.03	$\mathbf{0.1 \pm 0.01}$	NA	NA	NA
Elevators10000	0.07 ± 0.01	$\mathbf{0.01 \pm 0.01}$	NA	NA	NA
MotorCorrupt	2.04 ± 0.06	1.79 ± 0.05	1.73 ± 0.05	2.52 ± 0.09	$\mathbf{1.7 \pm 0.05}$
Boston	0.27 ± 0.02	$\mathbf{0.09 \pm 0.01}$	0.23 ± 0.02	0.19 ± 0.02	$\mathbf{0.09 \pm 0.02}$

Table 4.2 Results NLPD over 5 cross-validation folds with 10 replicates each. Models shown in comparison are sparse Gaussian (G), chained heteroscedastic Gaussian (CHG), Student-*t* Laplace approximation (Lt), Student-*t* variational approximation (Vt), and chained heteroscedastic Student-*t* (CHt).100 inducing points are used throughout.

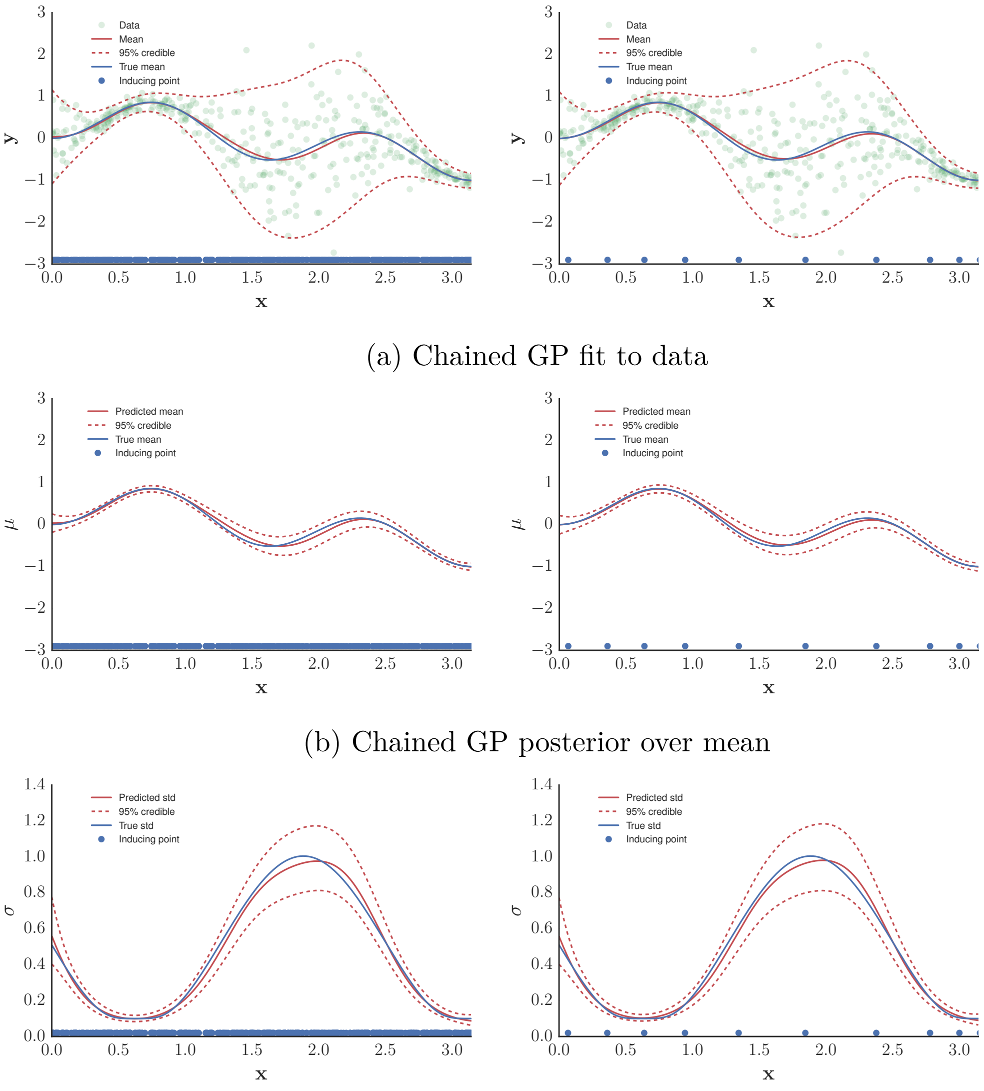
v_{f_i} denotes the ith diagonal element of the matrix $\boldsymbol{V}_f$. It may be possible in this Gaussian likelihood case to find the optimal $q(\boldsymbol{f})$ such that the bound collapses to that of Lazaro-Gredilla and Titsias (2011), however this would not allow for stochastic optimisation. The chained Gaussian process model provides a sparse extension, where a Gaussian distribution is assumed for the posterior over of $\boldsymbol{f}$, where as previously $q(\boldsymbol{f})$ had been collapsed out and could take any form. This sparse extension provides the ability to scale to much larger datasets whilst maintaining a similar variational lower bound.

We first use the chained GP approximation to show how the addition of input-dependent noise to a Gaussian process regression model affects performance, compared with a sparse Gaussian process model (Titsias, 2009). Performance is shown to improve as more data is provided as would be expected, making it clear that both models can scale with data, though the new model is more flexible when handling the distributions tails. A sparse Gaussian process with Gaussian likelihood is chosen in these experiments as a baseline, as a non-sparse Gaussian process cannot scale to the size of all the experiments.

The Elevator1000 uses a subset of 1000 randomly selected data-points from the Elevator dataset. In this data the heteroscedastic model (Chained GP) offers considerable improvement in terms of NLPD over the sparse GP (Table 4.2).

The second experiment with the Gaussian likelihood, Elevator10000, examines scaling of the model. Here a subset of 10000 data points of the Elevator dataset are used, and performance maintains an improvement as expected. Previous models for heteroscedastic Gaussian process models cannot scale, the chained GP model can encode the heteroscedastic setting and additionally scale to large datasets.

Following an example set by Cawley et al. (2004) that considers heteroscedastic kernel ridge regression, from which other publications have followed (Le et al., 2005),

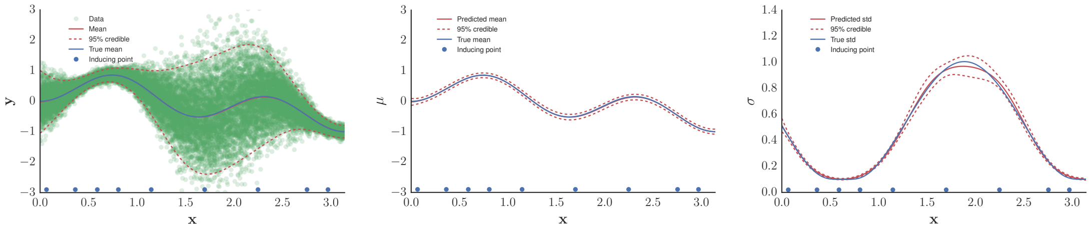


(a) Chained GP fit to data

(b) Chained GP posterior over mean

(c) Chained GP posterior over standard deviation

Fig. 4.6 a) The chained Gaussian process fit to a heteroscedastic dataset, the mean of the posterior for the conditional mean is shown alongside the 95% credible interval coming from the mean of the posterior for the conditional variance, both of these posteriors also have uncertainty associated with them however. b) The posterior mean with 95% credible intervals from the posterior. c) The posterior standard deviation, with 95% credible intervals from the posterior.

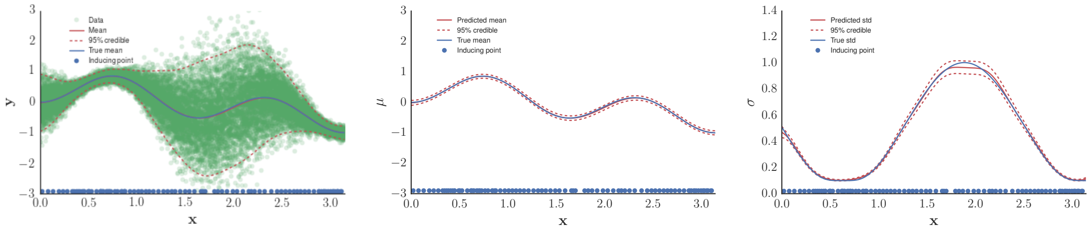


(a) Fit to data

(b) Posterior mean

(c) Posterior standard deviation

Fig. 4.7 Under a similar setting to Figure 4.6 the chained Gaussian process credible intervals collapse around the true values as the model becomes more certain in the presence of more data, particularly about the conditional standard deviation function



(a) Fit to data

(b) Posterior mean

(c) Posterior standard deviation

Fig. 4.8 Increasing the number of inducing points to 150 from 10 in the same context as Figure 4.7 reduces the predicted variance around the conditional variance, but it does not completely collapse around the true function

we also consider the accuracy of the chained GPs posterior approximation to the mean and variance functions. Cawley et al. (2004) provided a method that is able to obtain an estimate of the variance function with very little bias, an attractive property. The chained Gaussian process model was fit to two datasets, one consisting of 400 data points, the other 25000 data points. The univariate synthetic datasets contain heteroscedastic variance, identical to that used by Cawley et al. (2004). Inputs $\mathbf{x}$ are drawn from a uniform distribution in the interval $(0, \pi)$, and outputs $\mathbf{y}$ are drawn from a Gaussian distribution, the mean and variance of with are smooth functions of the input, $\mathbf{x}$,

$$\mathbf{x} \sim U(0, \pi),$$
$$\mathbf{y} \sim \mathcal{N}\left(\sin\left[\frac{5\mathbf{x}}{2}\right]\sin\left[\frac{3\mathbf{x}}{2}\right], \frac{1}{100} + \frac{1}{4}\left\{1 - \sin\left[\frac{5\mathbf{x}}{2}\right]\right\}^2\right)$$

Figures 4.6 and 4.7 show the posterior distribution over functions learnt for the mean and standard deviation, compared with the true mean and standard deviation. For the smaller dataset of 400 points, two chained Gaussian process fits were made, one with 400 inducing points, $\mathbf{z}$, fixed to the locations of $\mathbf{x}$; the other with 10 inducing points initialised uniformly across $(0, \pi)$; their locations were then subsequently optimised with respect to the lower bound along with the (hyper-) parameters of the model. For the larger dataset of 25000 data points, a model with 10 optimised inducing points was fit, as 25000 inducing points would be computationally infeasible.

Unlike the kernel ridge regression approach, that used a leave-one-out cross-validation to find a point estimate the conditional variance, the chained Gaussian process learns a posterior distribution over possible variance functions. The mean of the posterior over the conditional standard deviation ($\sqrt{e^g}$) is unfortunately is not exact for the chained Gaussian process model as illustrated in Figure 4.6c. However, the true conditional standard deviation function is easily contained within the 95% credibility intervals of the posterior prediction and the mean is not drastically away from the true function. Since the conditional variance function is so simple, the chained Gaussian process using only 10 inducing points almost exactly matches the model that uses all 400. Figure 4.7 shows the model fit to a much larger dataset of 25000 data points. In this case the posterior approximation to the variance function is significantly more certain, and collapses to an almost exact estimate of the true function. In this case we see that in the region of $x = 2$ there is relatively poor performance and only two inducing points located in this region. Figure 4.8 shows that by increasing the number of inducing points to 150 from 10, the model still contains relatively large

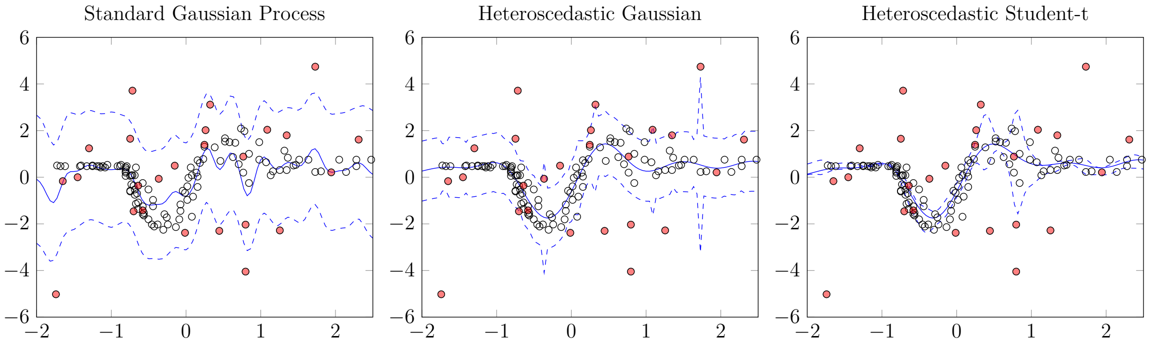


Fig. 4.9 Corrupted motorcycle dataset, fitted with a Gaussian process model with a Gaussian likelihood, a Gaussian process with input-dependent noise (heteroscedastic) with a Gaussian likelihood, and a Gaussian process with Student-t likelihood, with an input-dependent shape parameter. The mean is shown in solid and the variance is shown as dotted

predicted variance around the conditional variance in this region. This shows that when the model has few inducing points, Figure 4.7, it correctly focuses its attention on the region where predictive accuracy can be increased; where the conditional variance is low, $x = 0.5$. Unlike Le et al. (2005) our method is non-convex, and parameters may be optimised to local minima, however a measure of uncertainty around the functions is maintained.

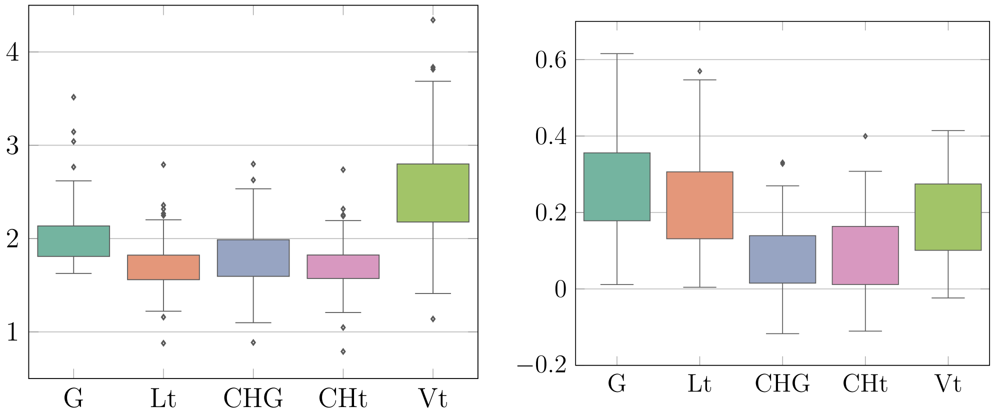
4.4.2 Robust Heteroscedastic Regression

We now investigate an extension of the Student-t likelihood that endows it with an input-dependent scale parameter. This is straightforward in the chained GP paradigm.

The corrupt motorcycle dataset is an artificial modification to the benchmark motorcycle dataset (Silverman, 1985) and shows the models capabilities more clearly. The original motorcycle dataset has had 25 of its data randomly corrupted with Gaussian noise of $\mathcal{N}(0, 3)$, simulating spurious accelerometer readings. We hope that our method will be robust and ignore such outlying values whilst capturing the trend in the underlying system. An input-dependent mean, μ, is set alongside an input-dependent scale that must be positive, σ. A constant degrees of freedom parameter ν, is initialised to 4.0 and then is optimised to its MAP solution.

$$y_i \sim St(\mu = f(\mathbf{x}_{i,:}), \sigma^2 = e^{g(\mathbf{x}_{i,:})}, \nu) \tag{4.10}$$

where $f(\mathbf{x}) = \text{GP}(\boldsymbol{\mu}_f, k_f(\mathbf{x}, \mathbf{x}'))$ and $g(\mathbf{x}) = \text{GP}(\boldsymbol{\mu}_g, k_g(\mathbf{x}, \mathbf{x}'))$. This provides a heteroscedastic extension to the Student-t likelihood. We compare the model



(a) NLPD motor

(b) NLPD Boston

Fig. 4.10 a) NLPD on corrupt motorcycle dataset. b) NLPD of Boston housing dataset. In NLPD lower is better, models shown in comparison are sparse Gaussian (G), Student-t Laplace approximation (Lt), Student-t variational approximation (Vt), chained heteroscedastic Gaussian (CHG), and chained heteroscedastic Student-t (CHt). Boxplots show the variation over 5 folds.

with a Gaussian process with homogeneous Student-t likelihood, approximated variationally (Hensman et al., 2015b) and with the Laplace approximation. Figure 4.9 shows the improved quality of the error bars with the chained heteroscedastic Student-t model. Learning a model with heavy tails allows outliers to be ignored, and so its input-dependent variance can be collapsed around the underlying function governing the variance, which in this case is known to be well modelled with a log Gaussian process (Goldberg et al., 1998; Lazaro-Gredilla and Titsias, 2011). It is also interesting to note the heteroscedastic Gaussian process models performance; although not able to completely ignore outliers the model has learnt a very short lengthscale and performs relatively well. This renders the prior over the scale parameter independent across the data, meaning that the resulting likelihood is more akin to a scale-mixture of Gaussian distributions (which endows appropriate robustness characteristics). The main difference is that the scale-mixture is based on a log-Gaussian prior, as opposed to the Student-t which is based on an inverse Gamma.

Figure 4.10 shows the NLPD on the corrupt motorcycle dataset and Boston housing dataset. The Boston housing dataset shows the median house prices throughout the Boston area, quantified by 506 data points, with 13 explanatory input variables (Kuß, 2006). We find that the chained heteroscedastic Gaussian process model already outperforms the Student-t model on this dataset, and the additional ability to use

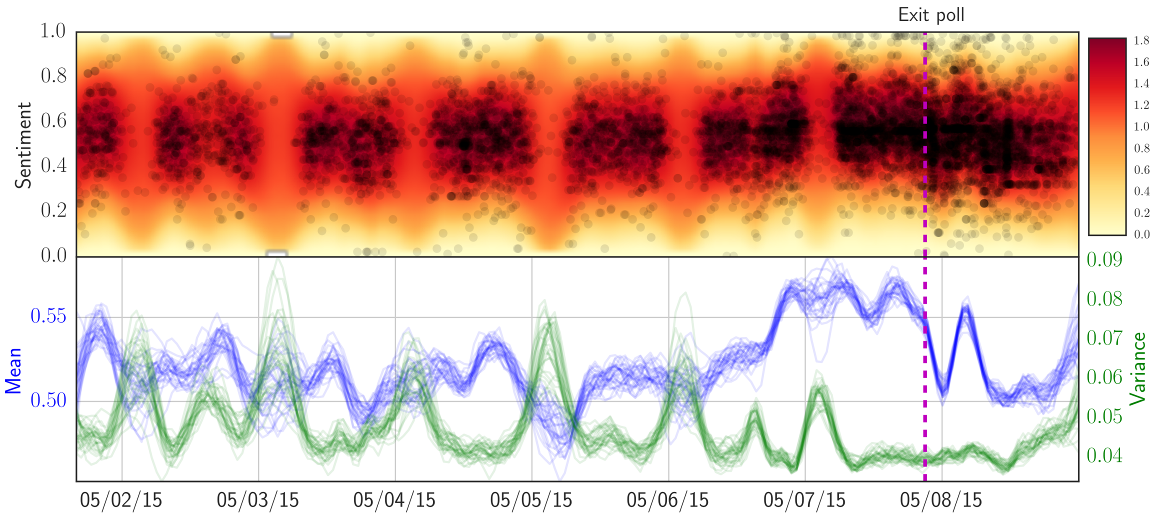


Fig. 4.11 Twitter sentiment from the UK general election modelled using a heteroscedastic beta distribution. The timing of the exit poll is marked and is followed by a night of tweets as election counts come in. Other night time periods have a reduced volume of tweets and a corresponding increase in sentiment variance. Ticks on the x-axis indicate midnight. The lower figure shows sample functions from the posterior of the mean and variance, in blue and green respectively.

heavier tails in the chained Student-t is not used. This ability to regress back to an already powerful model is a useful property of the chained Student-t model.

4.4.3 Twitter Sentiment Analysis in the UK Election

This section considers a model that further displays the adaptability of the model, on a novel dataset and with a novel heteroscedastic model. Here the sentiment during the run up to the UK 2015 general election is considered, focussing on tweets tagged as supporters of the Labour party. A sentiment analysis tagging system[2] was used to evaluate the positiveness of 105,396 tweets containing hashtags relating to recent the major political parties, in the run up to the election date. In this case, unlike several of the other datasets used within this Chapter, a sparse approximation is necessary as the dataset is too large for a more vanilla Gaussian process method.

The interest here is in modeling the distribution of positive sentiment as a function of time. The sentiment value is constrained to be between zero and one, and it is not necessarily the case that the distribution of tweets sentiment through time is unimodal. A natural likelihood to use in this case is the beta likelihood for beta regression. This

[2] Available from https://www.twinword.com/

allows bathtub shaped distributions to be accommodated, indicating tweets are either extremely positive or extremely negative, as well as unimodal distributions. The distribution of the tweets sentiment can be heterogenous throughout time by using Gaussian process models for each parameter of the beta distribution.

The Beta distribution traditionally is parameterised in the form, $y_i \sim B(\alpha, \beta)$ where $\alpha, \beta \in \mathbb{R}^+$ and observations $y_i \in (0,1)$, $\mathbf{x}_i \in \mathbb{R}^q$. An alternative parameterisation can be written in terms of the mean and dispersion of the distribution (Ferrari and Cribari-Neto, 2004),

$$p(y_i|\mu_i, \phi_i) = \frac{\Gamma(\phi_i)}{\Gamma(\mu_i\phi_i)\Gamma((1-\mu_i)\phi_i)} y_i^{\mu_i\phi_i - 1}(1-y_i)^{(1-\mu_i)\phi_i - 1}$$

where $0 < \mu_i < 1$ and $\phi_i > 0$. Since μ_i must be between zero and one, a natural transformation function is the logistic function. Since ϕ_i must be positive, it is additionally assigned a log GP prior,

$$y_i \sim B(\mu_i = \frac{e^{f(\mathbf{x}_{i,:})}}{1+e^{f(\mathbf{x}_{i,:})}}, \phi_i = e^{g(\mathbf{x}_{i,:})}), \tag{4.11}$$

where $f(\mathbf{x}) = \text{GP}\left(\boldsymbol{\mu}_f, k_f(\mathbf{x}, \mathbf{x}')\right)$ and $g(\mathbf{x}) = \text{GP}\left(\boldsymbol{\mu}_g, k_g(\mathbf{x}, \mathbf{x}')\right)$. This allows both the mean and variance, and hence shape of the beta distribution, change throughout time.

The justification for this re-parameterisation is that the mean and precision are likely to be more independent from one another than the α, β parameters, and the approximation proposed here assumes independence between the parameters. Experiments showed that although the resulting inferences were very similar, the mean and dispersion parameterisation exhibited faster convergence, likely due to the weaker true interactions between the parameters.

The upper section of Figure 4.11 shows the data and the probability of each sentiment value throughout time predicted by the model. The lower part shows samples of functions from the corresponding mean and variance posterior distributions induced by the above parameterisation. The 2015 general election was particularly interesting: polls throughout the election showed it to be a close race between the two major parties, Conservative and Labour. But at the end of polling an exit poll was released that predicted an outright win for the Conservatives. This exit poll proved accurate and is associated with a corresponding dip in the sentiment of the tweets. Other interesting aspects of the analysis include the reduction in number of tweets during the night and the corresponding increase in the variance of our estimates. For additional plots see Appendix C.3.

A similar model could be used over space as well as time; where the outputs are between 0 and 1. In an epidemiology setting this could be the proportion of a population with some characteristic of interest. For example studies may have been conducted throughout the country, and throughout time where the proportion of people in a sample with a given disease has been measured. In some cases a practitioner may wish to model the incidence rate based on the counts; in this case a Poisson model is appropriate as counts are integers and may be larger than one.

4.4.4 Decomposition of Poisson Processes

As discussed previously, the intensity of a Poisson process, $\lambda(x)$, can be trivially modelled as the product of two positive latent functions, $e^{f(\mathbf{x}_{i,:})}$ and $e^{g(\mathbf{x}_{i,:})}$. This corresponds to a generalised linear model (Section 2.3),

$$\begin{aligned}\log(\lambda) &= f(\mathbf{x}_{i,:}) + g(\mathbf{x}_{i,:})\\ \mathbf{y} &\sim \text{Poisson}(\lambda = e^{f(\mathbf{x}_{i,:})+g(\mathbf{x}_{i,:})} = e^{f(\mathbf{x}_{i,:})}e^{g(\mathbf{x}_{i,:})}),\end{aligned}$$

using a *log* link function.

Instead imagine forming a new process by combining two different underlying Poisson processes through addition, in this case both the intensities must be positive functions. The superposition property of Poissons means that the resulting process is also Poisson with intensity given by the sum of the underlying intensities.

To model this via a Gaussian process we have to assume that the intensity of the resulting Poisson, $\lambda(\mathbf{x})$ is a *sum* of two positive functions, $e^{f(\mathbf{x}_{i,:})}$ and $e^{g(\mathbf{x}_{i,:})}$ respectively,

$$\mathbf{y} \sim \text{Poisson}(\lambda = e^{f(\mathbf{x}_{i,:})} + e^{g(\mathbf{x}_{i,:})}), \tag{4.12}$$

where $f(\mathbf{x}) = \text{GP}\left(\boldsymbol{\mu}_f, k_f(\mathbf{x}, \mathbf{x}')\right)$ and $g(\mathbf{x}) = \text{GP}\left(\boldsymbol{\mu}_g, k_g(\mathbf{x}, \mathbf{x}')\right)$. There is no link function representation for this model, and so it takes the form of a chained-GP, unlike the product case above.

When the intensity function of a Poisson process is modelled by the log of a Gaussian process, the model is known as a Log-Gaussian Poisson process (Møller et al., 1998). Unfortunately inference for such a model is difficult as it requires an integral of the intensity function over a domain in which the counts are evaluated. If a log Gaussian process is used for the intensity function, such an integral is not analytically tractable. For low dimensional problems, this integral can be approximated on a grid, assuming that the Gaussian process is piecewise constant within the region (Hensman et al.,

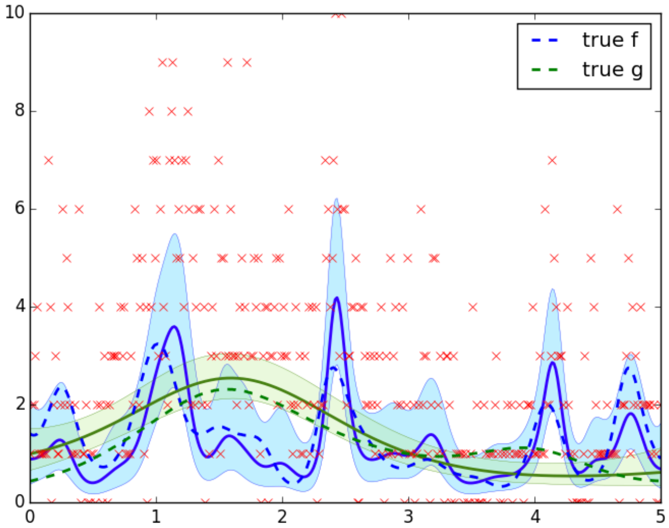


Fig. 4.12 Even with 350 data we can start to see the differentiation of the addition of a long lengthscale positive process and a short lengthscale positive process. Red crosses denote observations, dotted lines are the true latent functions generating the data using Eq (4.12), the solid line and associated error bars are the approximate posterior predictions, $q(\boldsymbol{f}^*), q(\boldsymbol{g}^*)$, of the latent processes.

2015a). This grid integration is very similar to the assumptions made for the piecewise constant proportional hazards survival likelihood discussed covered in Section 4.1.1. Indeed the hazard function can be seen as an intensity function, that additionally depends on the failure having not occurred in the past. This is the approach we will take, considering only two dimensional inputs regions. Progress has been made recently on a grid-free approach with variational approximation by Lloyd et al. (2015), using a squared transformation.

By discretising time and counting the number of failures within each time slice Efron (2002) considered a Poisson regression model for survival times; in this case only a linear predictor was used however. As the time slice window tends to 0, incidence counts will become either 0 or 1; in which case a Bernoulli likelihood can be used (Hanley and Miettinen, 2009). Focusing purely on the generative model of the data, the lack of a link function for Equation 4.12 does not present an issue. Figure 4.12 shows a simple demonstration of this idea on a simulated data set.

Using an additive model rather than a multiplicative model for the intensity function of counting process has been discussed previously in the context of linear models for survival analysis, with promising results Lin and Ying (1995).

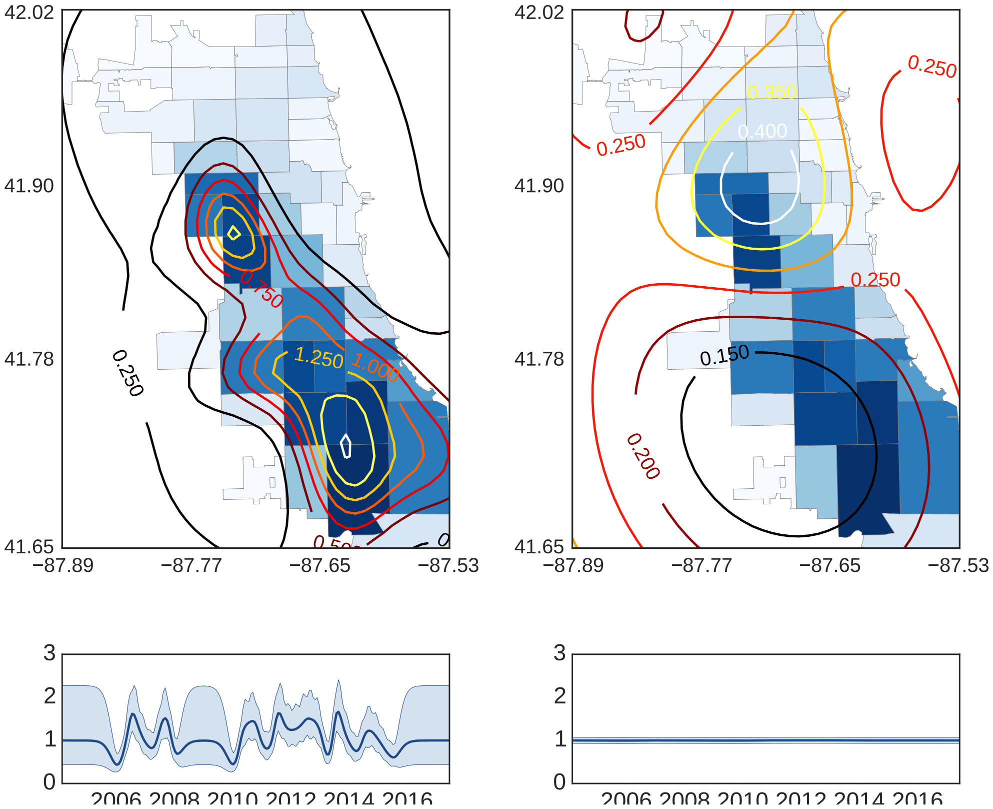


Fig. 4.13 Homicide rate maps for Chicago. The short length scale spatial process, $\lambda_1(x)$ (above-left) is multiplied in the model by a temporal process, $\mu_1(t)$ (below-left) which fluctuates with passing seasons. Contours of spatial process are plotted as deaths per month per zip code area. Error bars on temporal processes are at 5th and 95th percentile. The longer length scale spatial process, $\lambda_2(x)$ (above-right) has been modeled with little to no fluctuation temporally $\mu_2(t)$ (below-right).

To illustrate the model on real data, for failure intensities composed of multiple intensity functions, we considered homicide data in Chicago. Taking data from http://homicides.redeyechicago.com/ (see also Linderman and Adams (2014)) we aggregated data into three months periods by zip code. We considered an additive Poisson process with a particular structure for the covariance functions. We constructed a rate of the form:

$$\Lambda(\mathbf{x}, t) = \lambda_1(\mathbf{x})\mu_1(t) + \lambda_2(\mathbf{x})\mu_2(t)$$

where $\lambda_1(\mathbf{x}) = e^{f_1(\mathbf{x})}$, $\lambda_2(\mathbf{x}) = e^{g_1(\mathbf{x})}$, $\mu_1(t) = e^{f_2(t)}$ and $\mu_2(t) = e^{g_2(t)}$ where $f_1(\mathbf{x})$, $g_1(\mathbf{x})$ are spatial GPs and $f_2(t)$ and $f_2(t)$ are temporal GPs.

The overall rate decomposes into two separable rate functions, $\lambda_1(\mathbf{x})\mu_1(t)$ and $\lambda_2(\mathbf{x})\mu_2(t)$, but the overall rate function is not separable, as it is composed of a sum of two (positive) separable (Álvarez et al., 2012) rate functions. This structure allows us to decompose the homicide map into separate spatial maps that each evolve at different time rates. We selected one spatial map with a lengthscale of 0.04 and one spatial map with a lengthscale of 0.09. The time scales and variances of the temporal rate functions were optimised by maximum likelihood. The results are shown in Figure 4.13. The long lengthscale process hardly fluctuates across time, whereas the short lengthscale process, which represents more localised homicide activity, fluctuates across the seasons. This decomposition is possible and interpretable due to the structured underlying nature of the GPs inside the chained model.

4.4.5 Related Work

Standard Gaussian process regression assumes the mean function of a Gaussian likelihood can be modelled as a Gaussian process, and the noise variance β^{-1} is constant throughout the input range. In Section 4.4.1 we discussed one extension of this model, known as the heteroscedastic GP regression model (Goldberg et al., 1998), where the noise variance is dependent on the input. Since the noise variance must be positive, most approaches assign it a log GP prior, $y_i \sim \mathcal{N}\left(f(\mathbf{x}_{i,:}), e^{g(\mathbf{x}_{i,:})}\right)$, where $g(\mathbf{x}) = \mathrm{GP}\left(\boldsymbol{\mu}_g, k_g(\mathbf{x}, \mathbf{x}')\right)$. Unfortunately this results in an analytically intractable posterior distribution for $p(\boldsymbol{f}, \boldsymbol{g}|\mathbf{y})$ and so requires an approximation. Lazaro-Gredilla and Titsias (2011) propose a variational approximation in this case, though there are a number of other methods that perform inference in this type of model in a variety of other ways (Cawley et al., 2004; Kersting et al., 2007; Le et al., 2005; Snelson and Ghahramani, 2006b). The chained Gaussian process framework uses a similar variational approximation to Lazaro-Gredilla and Titsias (2011), however uses work from the sparse Gaussian process literature to scale this model. The chained Gaussian process model is most attractive in the situation where $n >> m$, as the number of variational parameters that must be learnt become more justifiable, optimisation of these parameters can be challenging when m is large.

Another generalisation of the standard GP regression with Gaussian likelihood assumption is to vary the scale of the process as a function of the inputs. Adams and Stegle (2008) suggest a log GP prior for the scale of the process giving rise to non-parametric non-stationarity in the model. Turner and Sahani (2011) took a related approach to develop probabilistic amplitude demodulation, here the amplitude (or scale) of the process was given by a Gaussian process with a link function given by

$\sigma = \log(\exp(\boldsymbol{f}) - 1)$. Finally Tolvanen et al. (2014) assign both the noise variance and the scale a log GP prior.

All of these variations on Gaussian process regression combine processes in a non-linear way with a Gaussian likelihood, but the chained Gaussian process framework focuses on systems that have non-Gaussian observation noise. This provides additional flexibility when specifying survival analysis models, as in Section 4.3.1, and relaxes assumptions made by existing models in the literature, Section 4.1. It also provides the capability to model a decomposed additive failure rate function for count data, this may be of interest within a epidemiology setting. Previous work in this domain of non-Gaussian likelihoods with multiple input-dependent parameters, include the use of the Laplace approximation (Vanhatalo et al., 2013), however this method scales poorly, $\mathcal{O}((bn)^3)$ and so isn't applicable to datasets of more modest sizes.

In other work (Nguyen and Bonilla, 2014) mixtures of Gaussian latent functions have also been applied for posterior distributions induced by non-Gaussian likelihoods, we expect such mixture distributions would also be applicable to our case. More recently this approach was extended (Dezfouli and Bonilla, 2015) to provide scalability utilising sparse methods similar to this work.

4.5 Conclusion

This chapter has discussed how the Gaussian processes framework can be used for survival analysis regression. Our novel contribution, the chained survival analysis model, gives the ability to relax two common assumptions made by the standard approaches. It allows the linearity assumption about the effect of the function of covariates made by many common survival models reviewed in Chapter 2 to be relaxed. It also provides the ability to allow multiple parameters of the survival likelihood to be input-dependent, allowing the shape of the hazard function to change in response to the individual; an assumption that is made to our knowledge by all accelerated failure time models. It achieves both of these goals whilst providing a scalable and flexible inference method that can be used for a number of other generative models that share similar assumptions, as shown by further experimental results. However, to carry out a survival analysis regression with this model, the covariates, $\mathbf{X}$, must be known, as is the case for most regressions. Chapters 5 and 6 consider what can be done when not all of the same input variables of all individuals have been measured. This is a common problem in clinical data and needs to be handled if correct inferences are to be made, one common is approach is missing data imputation followed by a subsequent regression.

5

Missing Data Imputation with Gaussian Processes

> *"The answer to any prediction problem is a probability distribution."*
>
> – Peter McCullogh via Peter Diggle

Clinical data is both complex in its nature, and often plagued by missingness of measurements. In practice the missing data must be handled in some principled way if subsequent inferences are to be made. There are number of popular ways in which missing data can be handled, many of which rely on the idea of *imputation* of the missing values. Imputed values will usually have some degree of uncertainty associated with them. Several popular methods for imputation, including principle component analysis (PCA), result in imputations that are normally distributed around a most probable value. Unfortunately clinical data often comes from a variety of sources, and so this normality assumption can be incorrect. The focus of this chapter will be on extending existing work on missing data imputation with PCA, to provide a more flexible model for the measurements, and to account for non-Gaussian distributed observations.

This chapter begins with an overview of mechanisms of missing data, to identify the applicability of the proposed model. Common existing approaches to handling of missing covariate data are then discussed. Our second major contribution will be then provided, a novel Bayesian missing data imputation method based on the *Bayesian Gaussian process latent variable model* (BGPLVM), known as the *Laplace approximation Bayesian Gaussian process latent variable model* (LABGPLVM). This method combines the variational approach to inference within the BGPLVM, with a Laplace approximation to handle non-Gaussian likelihood functions.

This extension will endow an already powerful latent variable model approach to imputation with the ability to handle non-Gaussian observations. To understand the complexities of the novel method, it will be necessary to review some of the models upon which it builds. During the derivation of the LABGPLVM, a number of previously unnoticed enlightening observations will come to light about the existing vanilla BGPLVM. Though the use of the vanilla BGPLVM as an imputation method has been considered previously, the model has not previously been published and there have been no studies of its effectiveness as an imputation method; as such this chapter also contains the first experimental results for this model.

5.1 Mechanisms of Missingness

The mechanisms by which data is missing has an impact on the methods that can be used to impute it. In this section a number of missingness mechanisms that are widely used are formalised and notation used for missing data throughout the chapter is defined.

Consider a matrix, $\mathbf{M}$, as a missing-data indicator matrix for a measurement matrix, $\mathbf{Y}$, such that $\mathbf{M}_{ij} = 1$ for all $\mathbf{Y}_{ij}$ that are missing, and $\mathbf{M}_{ij} = 0$ otherwise. Let $\mathbf{Y}^{\mathcal{O}}$ denote the observed elements of $\mathbf{Y}$, and $\mathbf{Y}^{\mathcal{M}}$ the missing elements. Note that we now use $\mathbf{Y}$ as the covariates that need to be imputed, in contrast to previous chapters where covariates were denoted $\mathbf{X}$. Regression models will be specified later to impute these partially missing observations and so this notation is appropriate as the predictor of a regression model.

Following Little and Rubin (1986), this chapter treats the missing-data indicators as random variables. The simplest mechanism of missingness is *missing completely at random* (MCAR). In this case the probability of being missing is completely independent of any measurements, as such the missingness is conditionally independent of $\mathbf{Y}$,

$$p(\mathbf{M}|\mathbf{Y}, \boldsymbol{\theta}_{\mathrm{m}}) = p(\mathbf{M}|\boldsymbol{\theta}_{\mathrm{m}}),$$

where $\boldsymbol{\theta}_{\mathrm{m}}$ denotes unknown parameters associated with the missingness mechanism. MCAR simply means that whether the value of $\mathbf{Y}$ is missing or not, does not depend on the measurement data, $\mathbf{Y}$, itself.

For MCAR data, it is typically valid to make subsequent inferences using the instances of data that have been fully observed, that is rows of $\mathbf{Y}$ that do not contain missing values, by simply ignoring any instances that contain missing data. When this is done, a model of the missingness mechanism itself is not required. This is called complete case analysis. However, if missingness is very common, dropping rows of $\mathbf{Y}$ that contain any missing values may leave little or no data to subsequently use in a regression model.

Another common and less restrictive mechanism is *missing at random* (MAR). In this case the probability that a measurement is missing, depends on observed measurements, $\mathbf{Y}^{\mathcal{O}}$, but not on the missing measurements, $\mathbf{Y}^{\mathcal{M}}$,

$$p(\mathbf{M}|\mathbf{Y}, \boldsymbol{\theta}_{\mathrm{m}}) = p(\mathbf{M}|\mathbf{Y}^{\mathcal{O}}, \boldsymbol{\theta}_{\mathrm{m}}). \tag{5.1}$$

An example of the MAR mechanism can be given as follows; consider a measurement of a patient that has been observed during a hospital visit; such as an positive indicator that the patient has been diagnosed with lung cancer. It is probable that the outcomes of unrelated tests, such as an eye-test, are missing as there are more urgent measurements to obtain from the patient. Conversely, it may be more probable that other tests related to the current diagnosis of lung cancer *have* been conducted than for a patient that has not received this diagnosis. In this case the probability that the data is missing, is dependant on other non-missing factors, $\mathbf{Y}^{\mathcal{O}}$, but not necessarily dependent on the missing values themself $\mathbf{Y}^{\mathcal{M}}$.

In the case of MAR data, knowledge of missing data values themselves, $\mathbf{Y}^{\mathcal{M}}$, is not required. A final mechanism is known as *missing not at random*, (MNAR or sometimes NMAR), and is the relaxation of MAR; whereby the missingness, is dependent not only on the observed values, but also the missing values, $\mathbf{Y}^{\mathcal{O}}$, that is

$$p(\mathbf{M}|\mathbf{Y}, \boldsymbol{\theta}_{\mathrm{m}}) = p(\mathbf{M}|\mathbf{Y}^{\mathcal{M}}, \boldsymbol{\theta}_{\mathrm{m}}) \qquad \text{or} \qquad p(\mathbf{M}|\mathbf{Y}, \boldsymbol{\theta}_{\mathrm{m}}) = p(\mathbf{M}|\mathbf{Y}^{\mathcal{M}}, \mathbf{Y}^{\mathcal{O}}, \boldsymbol{\theta}_{\mathrm{m}}).$$

In contrast to MCAR and MAR, MNAR requires an explicit model of $\mathbf{M}$, $p(\mathbf{M}|\boldsymbol{\theta}_{\mathrm{m}})$, to be provided or inferred to make inferences using the data, without introducing biases. In this research focuses on performing inferences within the context of MCAR and MAR data, one exception to this being censoring.

Censoring as discussed in Section 2.1.1 is an example of MNAR data, where the mechanism of missingness may depend on the underlying failure time, but where the mechanism is often understood (Little and Rubin, 1986). Consider for a moment censoring occurring solely at the end of a clinical trial, known as lost to follow-up. As in Chapter 2 let t_i denote the observed time of failure or censoring of a patient i, and ν_i denote the failure indicator, 1 if the failure happened, or 0 if censoring occurred. Let s_i denote the true (unobserved in the case $\nu_i = 0$) failure time. If the ν_i is dependent on s_i which is missing, the data can be considered MNAR. In the case that censoring occurs, $\nu = 0$, at the termination time, η, of a clinical trial that is shared amongst all patients,

$$\nu_i = \begin{cases} 0, & \text{if } s_i > \eta \\ 1, & \text{if } s_i \leq \eta. \end{cases}$$

The missing data mechanism for the failure indicator, ν, is then dependent on the value of the true but possibly missing failure data, s, and this information clearly needs to be taken into account when performing inferences. This is made clear by the fact that if a true failure time, s_i, is indeed missing, it is indicative that the patient has some attributes that encourage survival beyond the end of the trial, and so the failure indicator, ν_i is likely to be 0. Failure to include the failure times of individuals that were censored will bias the subsequent inference.

5.1.1 Ignorability

By treating the missingness indicators as random variables we must now consider what impact missingness mechanisms have on any Bayesian inference procedure that rely on the missing data.

Typically it is desirable to be able to calculate the marginal probability of the observed data, $\mathbf{Y}^{\mathcal{O}}$, given a model governed by parameters, $\boldsymbol{\theta}$, or the posterior probability of the parameters given the data. This may be for model comparison reasons, or for the purpose of optimisation and prediction. In the most general case our missingness mechanism can be given by $p(\mathbf{M}|\mathbf{Y}^{\mathcal{M}}, \mathbf{Y}^{\mathcal{O}}, \boldsymbol{\theta}_{\mathrm{m}})$. The marginal probability of the observed data, $\mathbf{Y}^{\mathcal{O}}$, and the missingness mechanism, $p(\mathbf{M}|\mathbf{Y}^{\mathcal{O}}, \mathbf{Y}^{\mathcal{M}}, \boldsymbol{\theta}_{\mathrm{m}})$, can be obtained by integrating out the missing data,

$$p(\mathbf{Y}^{\mathcal{O}}, \mathbf{M}|\boldsymbol{\theta}_{\mathrm{m}}, \boldsymbol{\theta}) = \int p(\mathbf{Y}^{\mathcal{O}}, \mathbf{Y}^{\mathcal{M}}|\boldsymbol{\theta}) p(\mathbf{M}|\mathbf{Y}^{\mathcal{O}}, \mathbf{Y}^{\mathcal{M}}, \boldsymbol{\theta}_{\mathrm{m}}) d\mathbf{Y}^{\mathcal{M}}.$$

The posterior distribution of the parameters $p(\boldsymbol{\theta}, \boldsymbol{\theta}_{\mathrm{m}}|\mathbf{Y}^{\mathcal{O}}, \mathbf{M})$ can then be given by combining a prior $p(\boldsymbol{\theta}, \boldsymbol{\theta}_{\mathrm{m}})$ and the full likelihood (Little and Rubin, 1986) as follows,

$$p(\boldsymbol{\theta}, \boldsymbol{\theta}_{\mathrm{m}}|\mathbf{Y}^{\mathcal{O}}, \mathbf{M}) \propto p(\mathbf{Y}^{\mathcal{O}}, \mathbf{M}|\boldsymbol{\theta}, \boldsymbol{\theta}_{\mathrm{m}}) p(\boldsymbol{\theta}, \boldsymbol{\theta}_{\mathrm{m}}).$$

In the case the missingness mechanism is MAR (Equation 5.1) the missingness is independent of the missing data. If we additionally assume that apriori the parameters defining the missingness mechanism, $\boldsymbol{\theta}_{\mathrm{m}}$, and the parameters of the model of the data itself, $\boldsymbol{\theta}$, are independent,

$$\begin{aligned} p(\boldsymbol{\theta}, \boldsymbol{\theta}_{\mathrm{m}} | \mathbf{Y}^{\mathcal{O}}, \mathbf{M}) &\propto \int p(\mathbf{Y}^{\mathcal{O}}, \mathbf{Y}^{\mathcal{M}} | \boldsymbol{\theta}) p(\mathbf{M} | \mathbf{Y}^{\mathcal{O}}, \mathbf{Y}^{\mathcal{M}}, \boldsymbol{\theta}_{\mathrm{m}}) p(\boldsymbol{\theta}, \boldsymbol{\theta}_{\mathrm{m}}) d\mathbf{Y}^{\mathcal{M}} \\ &= p(\mathbf{M} | \mathbf{Y}^{\mathcal{O}}, \boldsymbol{\theta}_{\mathrm{m}}) p(\boldsymbol{\theta}_{\mathrm{m}}) \times \int p(\mathbf{Y}^{\mathcal{O}}, \mathbf{Y}^{\mathcal{M}} | \boldsymbol{\theta}) p(\boldsymbol{\theta}) d\mathbf{Y}^{\mathcal{M}} \qquad (5.2) \\ &= \left[p(\mathbf{M} | \mathbf{Y}^{\mathcal{O}}, \boldsymbol{\theta}_{\mathrm{m}}) p(\boldsymbol{\theta}_{\mathrm{m}}) \right] \left[p(\mathbf{Y}^{\mathcal{O}} | \boldsymbol{\theta}) p(\boldsymbol{\theta}) \right]. \qquad (5.3) \end{aligned}$$

Since in the posterior, $\boldsymbol{\theta}$ and $\boldsymbol{\theta}_{\mathrm{m}}$, are independent, inference about the model parameters $\boldsymbol{\theta}$ can be made purely based on $p(\mathbf{Y}^{\mathcal{O}}|\boldsymbol{\theta})p(\boldsymbol{\theta})$, ignoring the missing data mechanism, $p(\mathbf{M}|\mathbf{Y}^{\mathcal{O}})$, entirely. Note that here we have made two assumptions, firstly that the data is MAR, and secondly that the parameters of the mechanism and the data are independent apriori. If either of these assumptions are invalid, the resulting inference is likely to be incorrect.

5.2 Imputation Methods

The most common way of handling missing data for subsequent inference is to impute their probable value, had it been observed. Unfortunately many of the simplest techniques are known to introduce biases. Although if the amount of missing data is relatively small compared to the size of the dataset, these biases may be small, but are clearly not desirable. Graham (2009) suggests that if missingness is less than 5% percent and completely random, complete case analysis on non-missing instances is acceptable. The most common imputation methods are single imputation methods (SI), and model based imputation methods.

SI includes simple methods such as using the mean of the non-missing values of a covariate (the mean of one column of $\mathbf{Y}^{\mathcal{O}}$). Another SI method is hot-deck (Andridge and Little, 2010), where observed values from similar instances are used, where a pool of similar instances may be collected with a k-nearest neighbour approach. A critical issue with SI methods is that they do not account for any of the uncertainty associated with the predictions made (Azur et al., 2011).

Model based imputation (MBI) methods specify a model for what the missing values should be. In this case it is natural to take into account the uncertainty of the imputed values in some way. A common approach to introducing this uncertainty is multiple imputation (MI) methods, that impute a number of possible imputed datasets. The widely adopted multivariate imputation by chained equations method (MICE) (Buuren and Groothuis-Oudshoorn, 1999), is a popular choice of MI method. MICE imputes a number of datasets, by introducing a series of simple regression models between covariates, in the form of a chain of equations. The resulting model does not necessarily result in a consistent model of the joint distribution of the missing data and non-missing data.

Subspace methods form another strain of imputation methods for missing data, the most popular of which being missing data imputation with PCA (Ilin and Raiko, 2010), and its probabilistic counterpart PPCA. The reader is directed to Graham (2009) for a more thorough review of existing imputation methods. The focus of the remainder of the chapter will be on extending existing Gaussian subspace model based approaches for missing data imputation, such that they can be applied to a broader range of problems. Principally this will be when it is known that the observations should not be normally distributed upon imputation, as will often be the case with clinical data.

In order to understand the complexities and assumptions of the more flexible novel model proposed in Section 5.6, as well as the limitations of existing popular subspace methods that are overcome, it will prove useful to begin from the more simplistic PPCA model that assumes both a linear latent mapping, and Gaussian observation noise assumption.

5.3 Probabilistic Principle Component Analysis

Probabilistic principle component analysis (Tipping and Bishop, 1999) (PPCA) is a simple latent variable model that is the probabilistic counterpart to PCA (Jolliffe, 1986). PCA attempts to find a lower dimensional representation of the data by finding the basis on which the variance of the data is maximised. It finds a linear mapping from some unknown latent space $\mathbf{X} = [\mathbf{x}_{1,:}, \mathbf{x}_{2,:}, \cdots, \mathbf{x}_{n,:}]^\top$ where $\mathbf{X} \in \mathbb{R}^{n \times q}$ to the observed data space $\mathbf{Y} = [\mathbf{y}_{1,:}, \mathbf{y}_{2,:}, \cdots, \mathbf{y}_{n,:}]^\top$ where $\mathbf{Y} \in \mathbb{R}^{n \times p}$. PPCA assumes that the observed data $\mathbf{Y}$ is then corrupted by Gaussian noise;

$$\mathbf{y}_{i,:} = g(\mathbf{x}_{i,:}) + \boldsymbol{\epsilon}_i,$$

The mapping function is assumed to be linear, $g(\mathbf{x}_{i,:}) = \mathbf{W}\mathbf{x}_{i,:}$. $\mathbf{W} \in \mathbb{R}^{p \times q}$ is a matrix of mapping weights between the latent points, $\mathbf{X}$, and the data space $\mathbf{Y}$. Note that unlike in previous chapters where p was assumed to be 1, multiple outputs are now taken into account. $\boldsymbol{\epsilon}_i$ is assumed to be i.i.d Gaussian noise,

$$p(\boldsymbol{\epsilon}_i|\beta^{-1}) = \mathcal{N}\left(\boldsymbol{\epsilon}_i|\mathbf{0}, \beta^{-1}\mathbf{I}\right).$$

Likelihood parameters will be collated as a vector, $\boldsymbol{\theta}_L$, for notational convenience later in this chapter; in the current state $\boldsymbol{\theta}_L = \{\beta^{-1}\}$.

The likelihood can then be written as,

$$p(\mathbf{Y}|\mathbf{X}, \mathbf{W}, \boldsymbol{\theta}_L) = \prod_{i=1}^{n} \mathcal{N}\left(\mathbf{y}_{i,:}|\mathbf{W}\mathbf{x}_{i,:}, \beta^{-1}\mathbf{I}\right). \tag{5.4}$$

Note that this is an identical setup to that of classical linear regression (Section 2.3), however in PPCA as well as the mapping weights, $\mathbf{W}$, not being provided, nor are the inputs $\mathbf{X}$. In PPCA, the latent locations of the training points $\mathbf{X}$ are treated as nuisance parameters as they are unknown. The linear mapping function itself is treated as the item of interest, and is to be inferred. The Bayesian approach to handling nuisance parameters is to assign the parameters a prior distribution, combine them with the likelihood, in this case Equation (5.4), and finally marginalise them out.

The prior distribution assigned for the nuisance parameters $\mathbf{X}$ is an i.i.d unit Gaussian that is independent across data, n,

$$p(\mathbf{X}) = \prod_{i=1}^{n} \mathcal{N}\left(\mathbf{x}_{i,:}|\mathbf{0}, \mathbf{I}\right). \tag{5.5}$$

To marginalise $\mathbf{X}$, first combine the prior with the likelihood $p(\mathbf{Y}|\mathbf{X}, \mathbf{W}, \beta^{-1})$ and then integrate over our latent points $\mathbf{X}$,

$$p(\mathbf{Y}|\mathbf{W}, \boldsymbol{\theta}_L) = \prod_{i=1}^{n} \int \mathcal{N}\left(\mathbf{y}_{i,:}|\mathbf{W}\mathbf{x}_{i,:}, \beta^{-1}\mathbf{I}\right) \mathcal{N}\left(\mathbf{x}_{i,:}|\mathbf{0}, \mathbf{I}\right) d\mathbf{x}_{i,:}. \tag{5.6}$$

An uncommon but useful attribute of the Gaussian distribution as used in Chapter 3 is that this integral is in fact analytically tractable (Appendix A.1),

$$p(\mathbf{Y}|\mathbf{W}, \boldsymbol{\theta}_L) = \prod_{i=1}^{n} \mathcal{N}\left(\mathbf{y}_{i,:}|\mathbf{0}, \mathbf{C}\right), \tag{5.7}$$

where $\mathbf{C} = \mathbf{W}\mathbf{W}^{\top} + \beta^{-1}\mathbf{I}$. This is then the marginal likelihood of the data with the latent points, $\mathbf{X}$ marginalised.

To find the maximum likelihood weights for the mapping, $\hat{\mathbf{W}}$, the marginal likelihood, Equation (5.7), is maximised with respect to the mapping weights, $\mathbf{W}$. Tipping and

Bishop (1999) show that finding the weights, $\hat{\mathbf{W}}$, that maximise the likelihood is equivalent to a eigenvalue problem, allowing especially efficient computation.

5.3.1 Dual Probabilistic Principle Component Analysis

An alternative, and in fact, equivalent approach to the above derivation of PPCA would be to instead marginalise the mapping weights, $\mathbf{W}$, this approach is known as dual PPCA (DPPCA) (Lawrence, 2004). This can be seen intuitively as instead treating the weights as a nuisance parameter; in this case the latent locations of the input $\mathbf{X}$, become the item of interest that is to be inferred.

The likelihood, Equation (5.4), is kept the same but a prior is assigned to the latent points, $\mathbf{X}$, rather than the weights, $\mathbf{W}$,

$$p(\mathbf{W}) = \prod_{i=1}^{n} \mathcal{N}\left(\mathbf{w}_{i,:}|\mathbf{0}, \mathbf{I}\right).$$

$\mathbf{W}$ can then be marginalise similarly to Equation (5.6),

$$\begin{aligned} p(\mathbf{Y}|\mathbf{X}, \boldsymbol{\theta}_L) &= \prod_{i=1}^{n} \int \mathcal{N}\left(\mathbf{y}_{i,:}|\mathbf{W}\mathbf{x}_{i,:}, \beta^{-1}\mathbf{I}\right) \mathcal{N}\left(\mathbf{w}_{i,:}|\mathbf{0}, \mathbf{I}\right) d\mathbf{w}_{i,:} \\ &= \prod_{i=1}^{n} \mathcal{N}\left(\mathbf{y}_{i,:}|\mathbf{0}, \mathbf{C}\right), \end{aligned}$$

where $\mathbf{C} = \mathbf{X}\mathbf{X}^\top + \beta^{-1}\mathbf{I}$. This model is equivalent to PPCA, however optimal locations of the latent points, $\mathbf{X}$, must now be found, instead of the optimal weights, $\mathbf{W}$. In this case the optimal locations can again be found by an eigenvalue problem (Tipping and Bishop, 1999).

5.4 Gaussian Process Latent Variable Model

The Gaussian process latent variable model (GPLVM) (Lawrence, 2004) is an extension of the dual PPCA model. Lawrence (2004) recognised that since the only place in the model that $\mathbf{X}$ is used, is inside an inner product, it is then possible to apply the *kernel trick* (Schölkopf, 2000). $\mathbf{C}$ can then be seen as an evaluation of the linear kernel (Section 3.1.1) plus some additional independent noise, $\mathbf{C} = \mathbf{X}\mathbf{X}^\top + \beta^{-1}\mathbf{I} = k(\mathbf{X}, \mathbf{X}) + \beta^{-1}\mathbf{I}$. This kernel function can then be replaced with any Mercer kernel (Schölkopf and Smola, 2002). By substituting in a different non-linear kernel function for k, this alleviates the linear restriction of the mapping $\mathbf{y}_{:,j} = g_j(\mathbf{x}_{i,:})$.

By rewriting the marginal likelihood,

$$\begin{aligned}
p(\mathbf{Y}|\mathbf{X}, \boldsymbol{\theta}_L, \boldsymbol{\theta}_K) &= \prod_{i=1}^{n} \mathcal{N}\left(\mathbf{y}_{i,:}|\mathbf{0}, \mathbf{C}\right) \\
&= \prod_{i=1}^{n} \frac{1}{(2\pi)^{\frac{p}{2}}} \frac{1}{|\mathbf{C}|^{\frac{1}{2}}} \exp(-\frac{1}{2}\mathbf{y}_{i,:}^{\top}\mathbf{C}^{-1}\mathbf{y}_{i,:}) \\
&= \frac{1}{(2\pi)^{\frac{np}{2}}} \frac{1}{|\mathbf{C}|^{\frac{1}{2}}} \exp(-\frac{1}{2}\mathbf{Y}^{\top}\mathbf{C}^{-1}\mathbf{Y}) \\
&= \prod_{j=1}^{p} \frac{1}{(2\pi)^{\frac{n}{2}}} \frac{1}{|\mathbf{C}|^{\frac{1}{2}}} \exp(-\frac{1}{2}\mathbf{y}_{:,j}^{\top}\mathbf{C}^{-1}\mathbf{y}_{:,j}), \qquad (5.8)
\end{aligned}$$

it is clear that the marginal likelihood is equivalent to the marginal likelihood of a product of p independent Gaussian processes as in Equation (3.7),

$$\begin{aligned}
p(\mathbf{Y}|\mathbf{X}, \boldsymbol{\theta}_L, \boldsymbol{\theta}_K) &= \prod_{j=1}^{p} \mathcal{N}\left(\mathbf{y}_{:,j}|\mathbf{0}, \mathbf{C}\right) \\
&= \prod_{j=1}^{p} \mathcal{N}\left(\mathbf{y}_{:,j}|\mathbf{0}, k(\mathbf{X}, \mathbf{X}) + \beta^{-1}\mathbf{I}\right) \\
&= \prod_{j=1}^{p} \int p(\mathbf{y}_{:,j}|\boldsymbol{f}_{:,j}, \boldsymbol{\theta}_L) p(\boldsymbol{f}_{:,j}|\mathbf{X}, \boldsymbol{\theta}_K) d\boldsymbol{f}_{:,j} \qquad (5.9)
\end{aligned}$$

Note that there *are* dependencies between the output dimensions; these dependencies are encoded within the kernel function, that all depend on the same set of inputs, $\mathbf{X}$, but given the latent function $\boldsymbol{f}_{:,j}$ the outputs dimensions are independent. The marginal likelihood is now also conditioned on the kernel hyper-parameters, $\boldsymbol{\theta}_K$ and as well as the likelihood parameters $\boldsymbol{\theta}_L$.

By using non-linear kernels in Equation (5.8), the GPLVM is obtained. The marginal likelihood is simply a product of p independent Gaussian process models, Section 3.1. In Section 3.1 the focus was purely on the case where the number of output dimensions, $p = 1$. From Equation 5.8 it is clear that for this model, when $p > 1$ the marginal likelihood factorises across output dimensions p. As such each output function, $\boldsymbol{f}_{:,j}$ is modelled by an independent Gaussian process from a *shared* input, $\mathbf{X}$. The observed data $\mathbf{y}_{:,j}$ is then a Gaussian corruption of these latent functions. The ability to use a non-linear mapping allows for substantially more complex dependencies between the output dimensions of the data than PPCA.

Unfortunately as a result of introducing a non-linear kernel, it is no longer possible to compute the ‘optimal’ maximum likelihood positions of the latent variables, $\mathbf{X}$, through a simple eigenvalue problem. To find these optimal positions the marginal likelihood, Equation (5.8), must be maximised with respect to $\mathbf{X}$ through optimisation, alongside the kernel parameters $\boldsymbol{\theta}_K$ and likelihood parameters, $\boldsymbol{\theta}_L$. This optimisation is non-convex and so can converge to sub-optimal local maxima. Additionally the marginal likelihood in Equation (5.8) is still conditioned on the latent locations, $\mathbf{X}$, and so only a maximum likelihood estimate of $\mathbf{X}$ is found, or a MAP solution if a prior is added for $p(\mathbf{X})$. In its current form it is not clear how to handle missing data.

5.4.1 Gaussian Process Latent Variable Model with Missing Data

In order to adapt the model to handle missing data we can use the properties outlined Section 5.1. The ignorability equality in Equation (5.2) showed that if the missingness mechanism for $\mathbf{M}$ is MAR or MCAR, it is valid to obtain the posterior distribution over latent parameters, $\boldsymbol{\theta}$, whilst not being concerned with the generative model that generated the missing values. Critically the evaluation of Equation (5.2) requires the ability to compute the following integral,

$$\int p(\mathbf{Y}^{\mathcal{O}}, \mathbf{Y}^{\mathcal{M}}|\boldsymbol{\theta})p(\boldsymbol{\theta})d\mathbf{Y}^{\mathcal{M}},$$

which is not possible for every type of model. In Equation (5.8) the marginal likelihood for the GPLVM was shown to factorise with respect to the dimensions. In the context of missing data in, $\mathbf{Y}$, it is possible to exploit this property by using the marginalisation properties of the Gaussian distribution (Appendix A.1.1). Consider a Gaussian distribution over a vector of random variables, $\mathbf{y} \sim \mathcal{N}(\boldsymbol{\mu}, \mathbf{C})$, then the marginal of the ith random variable is simply $y_i \sim \mathcal{N}(\mu_i, \mathbf{C}_{i,i})$. This can be applied to the above integral, with the GPLVM marginal likelihood,

$$\begin{aligned} p(\mathbf{Y}^{\mathcal{O}}|\mathbf{X}, \boldsymbol{\theta}_L, \boldsymbol{\theta}_K) &= \int \prod_{j=1}^{p} \mathcal{N}\left(\mathbf{y}_{:,j}|\mathbf{0}, k(\mathbf{X}, \mathbf{X}) + \beta^{-1}\mathbf{I}\right) d\mathbf{y}_{:,j}^{\mathcal{M}} \\ &= \prod_{j=1}^{p} \mathcal{N}\left(\mathbf{y}_{:,j}^{\mathcal{O}_j}|\mathbf{0}, k(\mathbf{X}^{\mathcal{O}_j}, \mathbf{X}^{\mathcal{O}_j}) + \beta^{-1}\mathbf{I}\right) \end{aligned}$$

where the notation $\mathbf{y}_{:,j}^{\mathcal{O}_j}$ represents only the n_j non-missing observations for data column $\mathbf{y}_{:,j}$. $\mathbf{X}^{\mathcal{O}_j}$ denotes the corresponding n_j inputs for these observations, note this contains all columns of $\mathbf{X}$ however. This marginalisation property has previously been exploited in the context of probabilistic matrix factorisation (Lawrence and Urtasun, 2009). Since the GPLVM is a generalisation of PPCA, a similar method can be used for imputation with PPCA. This property will also be used in a similar way for the novel model proposed in Section 5.6 to allow it to perform missing data imputation. Predictions for the GPLVM with a fixed certain input, $\mathbf{X}$, can be made using the standard GP prediction, Equation (3.8). Hernández-Lobato et al. (2014) have also proposed a related probabilistic matrix factorisation method that can also handle MNAR data, where the output variables, $\mathbf{Y}$, were ordinal.

5.5 Bayesian Gaussian Process Latent Variable Model

The Bayesian Gaussian process latent variable model (BGPLVM) (Titsias and Lawrence, 2010) attempts to rectify some of the shortcomings of the GPLVM; namely that since the latent inputs, $\mathbf{X}$, are optimised with respect to $p(\mathbf{Y}|\mathbf{X}, \boldsymbol{\theta}_L, \boldsymbol{\theta}_K)$, only maximum likelihood for these parameters can be sought. It is also possible to seek a MAP solution; by simply maximising $p(\mathbf{Y}|\mathbf{X})p(\mathbf{X})$ with respect to $\mathbf{X}$. However, this only encourages $\mathbf{X}$ to stay around $\mathbf{0}$, if a zero mean Gaussian prior is used, as in Equation (5.5). By not marginalizing out the latent variables, $\mathbf{X}$, and only optimizing $\mathbf{X}$ with respect to $p(\mathbf{Y}, \mathbf{X})$ the model is sensitive to overfitting (Damianou, 2015).

A fully Bayesian approach requires the computation of the true marginal likelihood, where $\mathbf{X}$ has been integrated out completely,

$$\begin{aligned} p(\mathbf{Y}|\boldsymbol{\theta}_L, \boldsymbol{\theta}_K) &= \int\int p(\mathbf{Y}|\mathbf{X}, \mathbf{W}, \boldsymbol{\theta}_L, \boldsymbol{\theta}_K)p(\mathbf{X})p(\boldsymbol{F})d\boldsymbol{F}d\mathbf{X} \\ &= \int p(\mathbf{Y}|\mathbf{X}, \boldsymbol{\theta}_L, \boldsymbol{\theta}_K)p(\mathbf{X})d\mathbf{X} \end{aligned} \tag{5.10}$$

The integral in the first line was carried out in Section 5.4; $p(\mathbf{Y}|\mathbf{X}, \boldsymbol{\theta}_L, \boldsymbol{\theta}_K)$ is given by the likelihood for the GPLVM, Equation (5.9).

Unfortunately $\mathbf{X}$ appears within the kernel function $k(\mathbf{X}, \mathbf{X})$, that is subsequently inverted in the evaluation of the GPLVM likelihood. Performing such an integration is analytically intractable.

The BGPLVM (Titsias and Lawrence, 2010) tackles this problem by forming variational lower bound on this marginal likelihood. The approach sidesteps the problem of the integration over the inverse operation, by introducing a set of inducing points,

$\mathbf{U} \in \mathbb{R}^{m\times p}$ at corresponding inducing inputs, $\mathbf{Z} \in \mathbb{R}^{m\times q}$, as covered in Section 3.3.1. A variational lower bound on the marginal likelihood $p(\mathbf{Y}|\boldsymbol{\theta}_L, \boldsymbol{\theta}_K)$ can then be found in a similar way to that of the sparse Gaussian process, Section 3.3; by finding the optimal form for $q(\mathbf{u}|\boldsymbol{\theta}_V)$ and plugging it into the bound, known as *collapsing*. The variational approximation also gives rise to an approximate posterior distribution $q(\mathbf{X}|\boldsymbol{\theta}_V) \approx p(\mathbf{X}|\mathbf{Y})$ that depends on variational parameters $\boldsymbol{\theta}_V$. A Gaussian mean-field approximation is used for this such that it is fully factorised as follows,

$$q(\mathbf{X}|\boldsymbol{\theta}_V) = \prod_{i=1}^{n} \mathcal{N}\left(\mathbf{x}_{i,:}|\boldsymbol{\mu}_{V\,i,:}, \boldsymbol{\Sigma}_{V\,i,i}\right)$$

Once optimised with respect to the lower bound on the marginal likelihood, this encodes the models belief in the location of the associated input data $\mathbf{X}$, given the observed data $\mathbf{Y}$, which is not simply a point estimate.

The derivation of the form of variational lower bound is involved and so only important details for the novel contributions are outlined; exhaustive details of the derivation can be found in Appendix D and in the following resources Damianou (2015); Damianou et al. (2016); Gal and van der Wilk (2014); Titsias and Lawrence (2010), though with slightly different nomenclature.

The resulting variational bound can be written in the form originally given by Titsias and Lawrence (2010),

$$\begin{aligned} L(q_{\mathbf{X}}) = \sum_{j=1}^{p} \left(\log \left[\frac{\beta^{\frac{n}{2}} \left|\mathbf{K}_{\mathbf{uu}}\right|^{\frac{1}{2}}}{(2\pi)^{\frac{n}{2}} \left|\beta\boldsymbol{\psi}_2 + \mathbf{K}_{\mathbf{uu}}\right|^{\frac{1}{2}}} e^{-\frac{1}{2}\mathbf{y}_{:,j}^{\top}\boldsymbol{D}\mathbf{y}_{:,j}} \right] - \frac{\beta\psi_0}{2} + \frac{\beta}{2}\mathrm{tr}\left(\mathbf{K}_{\mathbf{uu}}^{-1}\boldsymbol{\psi}_2\right) \right) \\ - \mathrm{KL}\left(q(\mathbf{X}|\boldsymbol{\theta}_V) \,\|\, p(\mathbf{X})\right), \end{aligned} \tag{5.11}$$

where

$$\boldsymbol{D} \triangleq \beta\mathbf{I} - \beta^2\boldsymbol{\psi}_1(\beta\boldsymbol{\psi}_2 + \mathbf{K}_{\mathbf{uu}})^{-1}\boldsymbol{\psi}_1{}^{\top} \tag{5.12}$$

$$\psi_0 \triangleq \mathbb{E}_{q(\mathbf{X}|\boldsymbol{\theta}_V)}\Big[\mathrm{tr}\left(k(\mathbf{X},\mathbf{X})\right)\Big] \in \mathbb{R}^{1\times 1} \tag{5.13}$$

$$\boldsymbol{\psi}_1 \triangleq \mathbb{E}_{q(\mathbf{X}|\boldsymbol{\theta}_V)}\Big[k(\mathbf{X},\mathbf{Z})\Big] \in \mathbb{R}^{n\times m} \tag{5.14}$$

$$\boldsymbol{\psi}_2 \triangleq \mathbb{E}_{q(\mathbf{X}|\boldsymbol{\theta}_V)}\left[\sum_{i=1}^{n} k(\mathbf{Z},\mathbf{X}_{i,:})k(\mathbf{X}_{i,:},\mathbf{Z})\right] \in \mathbb{R}^{m\times m}, \tag{5.15}$$

Note that $\psi_0, \boldsymbol{\psi}_1$ and $\boldsymbol{\psi}_2$ require the convolution of functions of kernels with the mean-field Gaussian distribution $q(\mathbf{X}|\boldsymbol{\theta}_V)$. Such convolutions are only analytically tractable for a subset of kernel functions, including the RBF kernel, ARD RBF kernel and the linear kernel. These quantities, and their closed form derivatives will also be used in the new non-Gaussian LABGPLVM, see Damianou (2015) for details on their form and numerical considerations. When confronted with a similar bound, Gal et al. (2015) choose to instead sample from $q(\mathbf{X}|\boldsymbol{\theta}_V)$, in which case any kernel function may be used.

As before, note that the bound on the marginal likelihood factorises across dimensions, p, and so the bound is a sum over the marginal log likelihood contribution for each dimension, $\log p(\mathbf{y}_{:,j}|\mathbf{Z})$. Also note that the variational bound $L(q_{\mathbf{X}})$ implicitly depends on the likelihood parameters, $\boldsymbol{\theta}_L$ and kernel hyper-parameters $\boldsymbol{\theta}_K$, as will all future marginal likelihoods and bounds beyond this point. These parameters must also be optimised alongside the variational parameters.

Although β^{-1} is defined as a scalar in the original model, and is shared amongst all outputs, p, this assumes that each output shares the same likelihood noise. This assumption is not necessary, although it reduces the computational complexity of the model, since $\boldsymbol{D}$ must only be computed once to compute the marginal likelihood contribution of all p outputs. This assumption can however be restrictive, as it assumes the same likelihood variance for each output, this would prove problematic when we consider the non-Gaussian likelihood extension of this model in Section 5.6.

5.5.1 Bayesian Gaussian Process Latent Variable Model with Missing Data

The BGPLVM bound, Equation (5.11) factorises across output dimensions, p. The bound however can also be rewritten in terms that sum over n (Dai et al., 2014; Gal et al., 2015). This means that the terms associated with missing data-points, can easily be marginalised. The reason why this is true will become more obvious upon the reformulation of the bound in Section 5.6.1. Intuitively, if the variational bound was derived considering a single output dimension, j, and the corresponding non-missing elements of $\mathbf{y}_{:,j}$ of $\mathbf{Y}$, a single element of the sum below would arise. Since the outputs are independent Gaussian processes, the bound becomes the sum of the bounds for each output dimension; considering only the relevant non-missing elements.

The bound can then be rewritten considering only the non-missing observations, $\mathbf{Y}^{\mathcal{O}}$,

$$
\begin{aligned}
L(q_{\mathbf{X}}) = \sum_{j=1}^{p} \Bigg(\log \left[\frac{\beta^{\frac{n_j}{2}} \left|\mathbf{K}_{\mathbf{uu}}\right|^{\frac{1}{2}}}{(2\pi)^{\frac{n_j}{2}} \left|\beta {\boldsymbol{\psi}_2}^{\mathcal{O}_j} + \mathbf{K}_{\mathbf{uu}}\right|^{\frac{1}{2}}} e^{-\frac{1}{2} {\mathbf{y}_{:,j}^{\mathcal{O}_j}}^{\top} \boldsymbol{D}^{\mathcal{O}_j} \mathbf{y}_{:,j}} \right] - \frac{\beta {\psi_0}^{\mathcal{O}_j}}{2} + \frac{\beta}{2} \mathrm{tr}\left(\mathbf{K}_{\mathbf{uu}}^{-1} {\boldsymbol{\psi}_2}^{\mathcal{O}_j} \right) \Bigg) \\
- \mathrm{KL}\left(q(\mathbf{X}|\boldsymbol{\theta}_V) \,\|\, p(\mathbf{X}) \right), \qquad (5.16)
\end{aligned}
$$

where the corresponding expectations depend only on the n_j, inputs $\mathbf{X}^{\mathcal{O}_j}$, that relate to the observations $\mathbf{y}_{:,j}^{\mathcal{O}_j}$,

$$\boldsymbol{D}^{\mathcal{O}_j} \triangleq \beta \mathbf{I} - \beta^2 {\boldsymbol{\psi}_1}^{\mathcal{O}_j} (\beta {\boldsymbol{\psi}_2}^{\mathcal{O}_j} + \mathbf{K}_{\mathbf{uu}})^{-1} {{\boldsymbol{\psi}_1}^{\mathcal{O}_j}}^{\top} \tag{5.17}$$

$${\psi_0}^{\mathcal{O}_j} \triangleq \mathbb{E}_{q(\mathbf{X}|\boldsymbol{\theta}_V)} \left[\operatorname{tr} \left(k(\mathbf{X}^{\mathcal{O}_j}, \mathbf{X}^{\mathcal{O}_j}) \right) \right] \in \mathbb{R}^{1 \times 1} \tag{5.18}$$

$${\boldsymbol{\psi}_1}^{\mathcal{O}_j} \triangleq \mathbb{E}_{q(\mathbf{X}|\boldsymbol{\theta}_V)} \left[k(\mathbf{X}^{\mathcal{O}_j}, \mathbf{Z}) \right] \in \mathbb{R}^{n_j \times m} \tag{5.19}$$

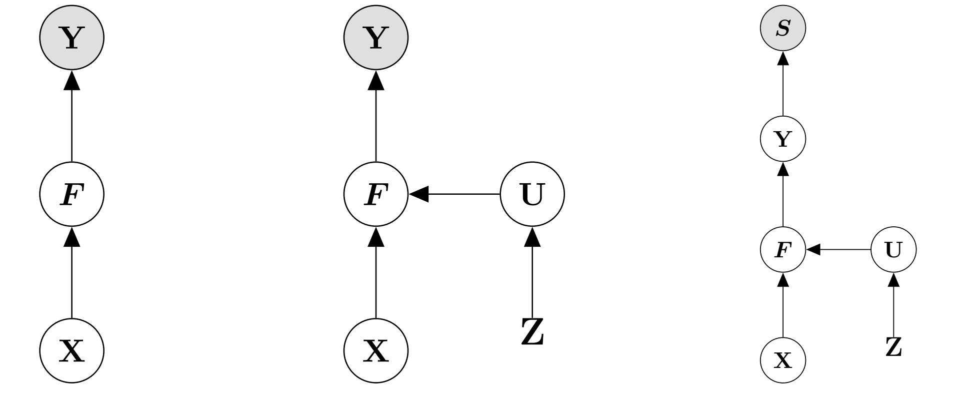
$${\boldsymbol{\psi}_2}^{\mathcal{O}_j} \triangleq \mathbb{E}_{q(\mathbf{X}|\boldsymbol{\theta}_V)} \left[\sum_{i=1}^{n_j} k(\mathbf{Z}, \mathbf{X}_{i,:}^{\mathcal{O}_j}) k(\mathbf{X}_{i,:}^{\mathcal{O}_j}, \mathbf{Z}) \right] \in \mathbb{R}^{m \times m}. \tag{5.20}$$

From Equation (5.16), it is clear that if the missingness pattern is different for each output dimension, j, the bound requires the matrix $\boldsymbol{D}^{\mathcal{O}_j}$ to be computed j times. This means that in the missing data scenario the optimisation of the bound will be approximately p times slower, which includes both an $\mathcal{O}(n_j m^2)$ operation, and a $\mathcal{O}(m^3)$ operation, as $\boldsymbol{D}^{\mathcal{O}_j}$ must be computed for each output dimension. This however means that no further computational complexity arises from assuming a different likelihood variance for each output dimension, β_j^{-1}. Since each output dimension, j, of the bound is independent, the computation of each outputs contribution could be computed in parallel. The development of this model was done in collaboration with Max Zwiessele and is based on unpublished work by Nicolo Fusi. In Section 5.6.5 we carry out the first systematic experiments evaluating the model's capabilities for imputation.

PPCA, GPVLM and BGPLVM models have been shown to be extremely effective at finding a lower dimensional representation of the data (Damianou et al., 2016; Ilin and Raiko, 2010; Lawrence, 2004), however unlike MICE, they all assume that all that the observations, $\mathbf{Y}^{\mathcal{O}}$, are a *Gaussian* corruption of the underlying latent function, $\boldsymbol{F}^{\mathcal{O}}$. Throughout the previous chapters it has become clear that such an assumption is unrealistic, particularly when dealing with a range of inputs as diverse as those collected in clinical trials, including survival times.

5.6 Bayesian GPLVM for Non-Gaussian Likelihoods

This provokes the question whether it is possible to relax this normality assumption during imputation, whilst maintaining the advantages of the BGPLVM. We now consider a novel model that relaxes the assumption of a Gaussian likelihood in the BGPLVM. This will provide a basis for a number of applications; including



(a) GPLVM (b) BGPLVM (c) LABGPLVM

Fig. 5.1 Graphical models of the Gaussian process latent variable models introduced in this chapter

dimensionality reduction, and imputation, and in the subsequent chapter, regression; all with uncertain inputs and non-Gaussian likelihoods. A similar model was put forward by Andrade (2015), using expectation propagation; without considering its missing data counterpart and so not realising its advantages in the area of imputation. Although expectation propagation is effective for classification (Kuss and Rasmussen, 2005; Nickisch and Rasmussen, 2008), it often has convergence difficulties, and is a computationally expensive method of inference. Each likelihood implemented also requires a relatively complex derivation of corresponding site terms. We would like to have the additional flexibility of easily choosing from a range likelihoods, where different likelihoods can be used for each output dimension. This allows for the array of data-types we may encounter in clinical data to be easily incorporated. We propose using the Laplace approximation, Section 3.2.1, to the variational bound of the BGPLVM to remedy this short coming of the BGPLVM. This new model is known as the *Laplace approximation Bayesian Gaussian process latent variable model* (LABGPLVM). Unlike previous works, we will focus on the task of missing data imputation, with the aim of applying it to real clinical data in the future work.

The approach to inference in this model requires an additional assumption to made in order to aid computation. The observations, that will now be denoted $\boldsymbol{S}$, are assumed to be a non-Gaussian corruption of a now *latent function* $\mathbf{Y}$. For the remainder of this chapter $\mathbf{Y}$ will become an underlying latent function though still a Gaussian corruption of $\boldsymbol{F}$ unless otherwise specified, and $\boldsymbol{S}$ will be the non-normally distributed observed variables. To be clear, $p(\mathbf{Y}|\mathbf{Z}, \boldsymbol{\theta}_L, \boldsymbol{\theta}_K)$ will be the same as in preceding sections, however $\mathbf{Y}$ is now assumed to be latent, not observed. The observations are then a non-Gaussian corruption of the latent $\mathbf{Y}$, $p(\boldsymbol{S}|\mathbf{Y})$. Graphical models showing the GPLVM, BGPLVM and our LABGPLVM model can be found in Figure 5.1. By making this additional assumption, the latent function $\mathbf{Y}$ is now equivalent to the latent function $\boldsymbol{F}$ as standard Gaussian process regression, however now a usually small amount of additional independent Gaussian noise, β^{-1} is added to each output dimension. This additional assumption will aid inference of the marginal likelihood of the LABGPLVM model, whilst not overly impairing the model with unrealistic assumptions. As in the BGPLVM, this noise variance can either be shared between all output dimensions j or each output may contain its own, β_j^{-1}.

The main focus of the model is to provide the ability to model non-Gaussian likelihoods. The model essentially uses the BGPLVM as a Gaussian prior, and now allows a non-Gaussian likelihood to be used for the observations $\boldsymbol{S}$. To compute the marginal likelihood of the model, first we must obtain a Gaussian form for the marginal likelihood of the BGPLVM, $p(\mathbf{Y}|\mathbf{Z}, \boldsymbol{\theta}_L, \boldsymbol{\theta}_K)$, and then we must perform a subsequent integration over the non-Gaussian likelihood, $p(\boldsymbol{S}|\mathbf{Y})$. In practice as we have seen in Section 5.5 that exact computation of $p(\mathbf{Y}|\mathbf{Z}, \boldsymbol{\theta}_L, \boldsymbol{\theta}_K)$ for the BGPLVM is not tractable, and we must use the variational bound as a surrogate.

The additional integral required to calculate the approximate bound on the marginal likelihood will take the form,

$$L(q_{\mathbf{X}})_{\text{lap}} = \log \int p(\boldsymbol{S}|\mathbf{Y}) \exp(L(q_{\mathbf{X}})) d\mathbf{Y} \tag{5.21}$$

In Section 3.2.1 the Laplace approximation was introduced for Gaussian process regression with non-Gaussian likelihoods. To recap, the marginal likelihood is the integral of a *Gaussian* prior, $p(\boldsymbol{f})$ and a *non-Gaussian* likelihood $p(\mathbf{y}|\boldsymbol{f})$. For the LABGPLVM, a reformulation of the variational lower bound of the BGPLVM is required such that it can be treated as a Gaussian prior with respect to $\mathbf{Y}$, $p(\mathbf{Y}|\mathbf{Z}, \boldsymbol{\theta}_L, \boldsymbol{\theta}_K) \approx \exp(L(q_{\mathbf{X}}))$. In doing so we will be able to compute an approximation to the integral in Equation (5.21) with a non-Gaussian likelihood $p(\boldsymbol{S}|\mathbf{Y})$ to account for the non-normality of the observed data $\boldsymbol{S}$.

We will begin by manipulating the original form of the BGPLVM bound, Equation (5.11), to a form more easily applicable to the Laplace approximation. By reformulating the bound, a novel interpretation of how the BGPLVM bound directly relates to the sparse Gaussian processes, Section 3.3 is found. These additional insights about the model as a result of this reformulation are discussed in Section 5.6.1. The Laplace approximation will then be applied to the reformulated bound, providing inference capabilities for both the LABGPLVM, and its missing data counterpart. The first set of experiments comparing the missing data BGPLVM, missing data PCA, and

MICE models in the context of imputation are then carried out. A comparison with the novel LABGPLVM method is then provided, showing the strengths and weaknesses of each approach to imputation.

5.6.1 Reformulating Bayesian Gaussian Process Latent Variable Model Bound

In its current representation the bound of the BGPLVM, Equation (5.11), appears to have an exponential quadratic form, that in turn implies normality with precision matrix $\boldsymbol{D}$, however the structure of the completed Gaussian distribution in $\mathbf{y}_{:,j}$ is not immediately clear. It will be convenient to find the associated Gaussian with respect to $\mathbf{y}_{:,j}$, to allow the Laplace approximation to be applied with little modification. This can be done by completing the square with respect to $\mathbf{y}_{:,j}$, that will then leave some extra terms that are independent of the precise values of $\mathbf{y}_{:,j}$. A full derivation of the below result can be found in the Appendix D.2.3.

Firstly, consider the inverse of the apparent precision matrix, $\hat{\mathbf{K}} \triangleq \boldsymbol{D}^{-1}$, which is the covariance matrix of $p(\mathbf{y}) = \mathcal{N}\left(\mathbf{y}|\mathbf{0}, \hat{\mathbf{K}}\right)$. By applying the matrix inversion lemma (Appendix A.2.1) in reverse to $\boldsymbol{D}^{-1}$, we can obtain the corresponding covariance matrix $\hat{\mathbf{K}}$,

$$
\begin{aligned}
(\beta\mathbf{I} - \beta\boldsymbol{\psi}_1(\beta\boldsymbol{\psi}_2 + \mathbf{K}_{\mathbf{uu}})^{-1}\boldsymbol{\psi}_1{}^{\top})^{-1} &= (\underbrace{\beta\mathbf{I}}_{A} + \underbrace{(-\beta\mathbf{I})^{\top}\boldsymbol{\psi}_1}_{U}\underbrace{(\beta\boldsymbol{\psi}_2 + \mathbf{K}_{\mathbf{uu}})^{-1}}_{B}\underbrace{\boldsymbol{\psi}_1{}^{\top}(\beta\mathbf{I})}_{V})^{-1} \\
&= \beta^{-1}\mathbf{I} + \boldsymbol{\psi}_1(\mathbf{K}_{\mathbf{uu}} + \beta(\boldsymbol{\psi}_2 - \boldsymbol{\psi}_1{}^{\top}\boldsymbol{\psi}_1)^{-1}\boldsymbol{\psi}_1{}^{\top} \triangleq \hat{\mathbf{K}}.
\end{aligned}
\tag{5.22}
$$

Using this form of the covariance matrix, and completing the square of the original bound, Equation (5.11), gives

$$
\begin{aligned}
L(q_{\mathbf{X}}) = \sum_{j=1}^{p}\Bigg(&\log\mathcal{N}\left(\mathbf{y}_{:,j}|\mathbf{0}, \hat{\mathbf{K}}\right) - \frac{1}{2}\log\left|\beta^{-1}\mathbf{I}\right| - \frac{1}{2}\log\left|\mathbf{K}_{\mathbf{uu}}^{-1}\right| - \frac{1}{2}\log\left|\beta\boldsymbol{\psi}_2 + \mathbf{K}_{\mathbf{uu}}\right| \\
&-\frac{\beta}{2}\psi_0 + \frac{\beta}{2}\mathrm{tr}\left(\mathbf{K}_{\mathbf{uu}}^{-1}\boldsymbol{\psi}_2\right) + \frac{1}{2}\log\left|\hat{\mathbf{K}}\right|\Bigg) - \mathrm{KL}\left(q(\mathbf{X}|\boldsymbol{\theta}_V)\,\|\,p(\mathbf{X})\right)
\end{aligned}
$$

where additional determinant terms have been introduced to complete the square.

The determinants can be further simplified using the matrix determinant lemma (Appendix A.2.2),

$$
\begin{aligned}
&-\frac{1}{2}\log\left|\beta^{-1}\mathbf{I}\right| - \frac{1}{2}\log\left|\mathbf{K}_{\mathbf{uu}}^{-1}\right| - \frac{1}{2}\log\left|\beta\boldsymbol{\psi}_2 + \mathbf{K}_{\mathbf{uu}}\right| \\
&\qquad + \frac{1}{2}\log\left|\underbrace{\beta^{-1}\mathbf{I}}_{A} + \underbrace{\boldsymbol{\psi}_1}_{U}\underbrace{\left(\mathbf{K}_{\mathbf{uu}} + \beta(\boldsymbol{\psi}_2 - {\boldsymbol{\psi}_1}^{\top}\boldsymbol{\psi}_1)^{-1}\right.}_{W}\underbrace{{\boldsymbol{\psi}_1}^{\top}}_{V^{\top}}\right| \\
&= -\frac{1}{2}\log\left|\mathbf{K}_{\mathbf{uu}} + \beta(\boldsymbol{\psi}_2 - {\boldsymbol{\psi}_1}^{\top}\boldsymbol{\psi}_1)\right| - \frac{1}{2}\log\left|\mathbf{K}_{\mathbf{uu}}^{-1}\right|,
\end{aligned}
$$

giving the following form of the variational bound, where the square has been completed with respect to $\mathbf{y}_{:,j}$,

$$
\begin{aligned}
L(q_{\mathbf{X}}) = \sum_{j=1}^{p}\Bigg(&\log\mathcal{N}\left(\mathbf{y}_{:,j}|\mathbf{0},\hat{\mathbf{K}}\right) - \frac{1}{2}\log\left|\mathbf{K}_{\mathbf{uu}} + \beta(\boldsymbol{\psi}_2 - {\boldsymbol{\psi}_1}^{\top}\boldsymbol{\psi}_1)\right| - \frac{1}{2}\log\left|\mathbf{K}_{\mathbf{uu}}^{-1}\right| \\
&-\frac{\beta\psi_0}{2} + \frac{\beta}{2}\mathrm{tr}\left(\mathbf{K}_{\mathbf{uu}}^{-1}\boldsymbol{\psi}_2\right)\Bigg) - \mathrm{KL}\left(q(\mathbf{X}|\boldsymbol{\theta}_V)\,\|\,p(\mathbf{X})\right). \qquad (5.23)
\end{aligned}
$$

Importantly this form of the bound makes it clear that the variational lower bound is a series of Gaussian processes, one for each output, as in Equation (5.9); however in this model there are a number of terms that regularize various aspects of the model, and there is a specific structure of covariance. It also makes it clear that the variational bound as a whole is a Gaussian distribution with respect to $\mathbf{Y}$, as the product of Gaussian distributions is a Gaussian itself, this will allow us to apply the Laplace approximation with relative ease.

5.6.2 Bayesian Gaussian Process Latent Variable Model Insights

There are additional insights to be had from studying this reformulation of the lower bound that are not made clear by the form provided in the original form of the bound Equation (5.11).

The reformulation makes transparent the models relationship to the variational sparse Gaussian process, Section 3.3. In the variational sparse Gaussian process lower bound, the form the marginal covariance is $\mathbf{K} = \beta^{-1}\mathbf{I} + \mathbf{K}_{\boldsymbol{f}\mathbf{u}}\mathbf{K}_{\mathbf{uu}}^{-1}\mathbf{K}_{\mathbf{u}\boldsymbol{f}}$. In the reformulated BGPLVM bound, Equation (5.23), the covariance is then given by $\hat{\mathbf{K}} = \beta^{-1}\mathbf{I} + \boldsymbol{\psi}_1(\mathbf{K}_{\mathbf{uu}} + \beta(\boldsymbol{\psi}_2 - {\boldsymbol{\psi}_1}^{\top}\boldsymbol{\psi}_1))^{-1}{\boldsymbol{\psi}_1}^{\top}$.

From the form of $\boldsymbol{\psi}_1$, Equation (5.14), it is clear that it is the analog to $\mathbf{K}_{\boldsymbol{f}\mathbf{u}}$, where the uncertainty of the input $\mathbf{X}$ is integrated out. However in the BGPLVM bound, the $\mathbf{K}_{\mathbf{uu}}^{-1}$, covariance term of the sparse Gaussian process is replaced with $(\mathbf{K}_{\mathbf{uu}} + \beta(\boldsymbol{\psi}_2 - \boldsymbol{\psi}_1{}^{\top}\boldsymbol{\psi}_1))^{-1}$.

From the definitions of $\boldsymbol{\psi}_2$ and $\boldsymbol{\psi}_1$, Equations (5.14) and (5.15) respectively,

$$\boldsymbol{\psi}_2 - \boldsymbol{\psi}_1{}^{\top}\boldsymbol{\psi}_1 = \mathbb{E}_{q(\mathbf{X}|\boldsymbol{\theta}_V)}\Big[\mathbf{K}_{\mathbf{u}\boldsymbol{f}}\mathbf{K}_{\boldsymbol{f}\mathbf{u}}\Big] - \mathbb{E}_{q(\mathbf{X}|\boldsymbol{\theta}_V)}\Big[\mathbf{K}_{\boldsymbol{f}\mathbf{u}}\Big]\mathbb{E}_{q(\mathbf{X}|\boldsymbol{\theta}_V)}\Big[\mathbf{K}_{\mathbf{u}\boldsymbol{f}}\Big]. \tag{5.24}$$

This quantity is the covariance of $\mathbf{K}_{\boldsymbol{f}\mathbf{u}}$ and $\mathbf{K}_{\mathbf{u}\boldsymbol{f}}$, and so this term is zero when $q(\mathbf{X}|\boldsymbol{\theta}_V)$ is a delta function. Informally the delta function can be viewed as the limit of a Gaussian distribution where the variance shrinks to zero.

In Equation (5.24) if the model is completely certain about the location of the input $\mathbf{X}$, the variance of $q(\mathbf{X}|\boldsymbol{\theta}_V)$ will shrink. $\mathbb{E}_{q(\mathbf{X}|\boldsymbol{\theta}_V)}\Big[\mathbf{K}_{\boldsymbol{f}\mathbf{u}}\Big]$ then in turn reverts back to $\mathbf{K}_{\boldsymbol{f}\mathbf{u}}$, and similarly so for $\mathbf{K}_{\mathbf{u}\boldsymbol{f}}$. $\mathbb{E}_{q(\mathbf{X}|\boldsymbol{\theta}_V)}\Big[\mathbf{K}_{\mathbf{u}\boldsymbol{f}}\mathbf{K}_{\boldsymbol{f}\mathbf{u}}\Big]$ will also become to $\mathbf{K}_{\mathbf{u}\boldsymbol{f}}\mathbf{K}_{\boldsymbol{f}\mathbf{u}}$.

Finally, $\hat{\mathbf{K}}$ then becomes $\beta^{-1}\mathbf{I} + \mathbf{K}_{\boldsymbol{f}\mathbf{u}}\mathbf{K}_{\mathbf{uu}}^{-1}\mathbf{K}_{\mathbf{u}\boldsymbol{f}}$; which is identical to the standard sparse GP model. The extra determinant terms would also cancel. This makes it clear that the BGPLVM is simply a series of variational sparse Gaussian processes, where uncertainty of the input location, $\mathbf{X} \sim q(\mathbf{X}|\boldsymbol{\theta}_V)$ is taken into account.

5.6.3 Laplace Approximation

From the generative modelling perspective, the LABGPLVM approximation to the variational lower bound of the marginal likelihood can now be written as follows, using the reformulated BGPLVM variational bound as a surrogate for $p(\mathbf{Y}|\boldsymbol{\theta}_L, \boldsymbol{\theta}_K, \mathbf{Z})$,

$$\begin{aligned} p(\boldsymbol{S}|\mathbf{Z}, \boldsymbol{\theta}_L, \boldsymbol{\theta}_K) &= \int p(\boldsymbol{S}|\mathbf{Y}, \boldsymbol{\theta}_L) p(\mathbf{Y}|\boldsymbol{\theta}_K, \boldsymbol{\theta}_L, \mathbf{Z}) d\mathbf{Y} \\ &\geq \int p(\boldsymbol{S}|\mathbf{Y}, \boldsymbol{\theta}_L) \exp(L(q_{\mathbf{X}})) d\mathbf{Y} \qquad (5.25) \\ &\triangleq \exp(L(q_{\mathbf{X}})_{\text{lap}}). \qquad (5.26) \end{aligned}$$

In the last line an approximation is made to define $\exp(L(q_{\mathbf{X}})_{\text{lap}})$ as the integral in Equation (5.25) is intractable. This intractability arises as, even though $\exp(L(q_{\mathbf{X}}))$ is now Gaussian in $\mathbf{Y}$, the likelihood, $p(\boldsymbol{S}|\mathbf{Y}, \boldsymbol{\theta}_L, \boldsymbol{\theta}_K)$ is now assumed to be non-Gaussian. The approximation we will use, is the Laplace approximation, although a range of other approximation methods may be used, this will be discussed in the conclusion.

The full approximate bound follows from the reformulated BGPLVM bound, Equation (5.23) and the required integral (5.25),

$$
\begin{aligned}
L(q_{\mathbf{X}})_{\text{lap}} &= \log \int p(\boldsymbol{S}|\mathbf{Y}, \boldsymbol{\theta}_L) \exp(L(q_{\mathbf{X}})) d\mathbf{Y} \\
&= \log \int p(\boldsymbol{S}|\mathbf{Y}, \boldsymbol{\theta}_L) \exp \Bigg[\sum_{j=1}^{p} \Bigg(\log \mathcal{N}\left(\mathbf{y}_{:,j}|\mathbf{0}, \hat{\mathbf{K}}_j\right) - \frac{1}{2} \log \left|\mathbf{K}_{\mathbf{uu}} + \beta(\boldsymbol{\psi}_2 - {\boldsymbol{\psi}_1}^{\top}\boldsymbol{\psi}_1)\right| \\
&\qquad - \frac{1}{2} \log \left|\mathbf{K}_{\mathbf{uu}}^{-1}\right| - \frac{\beta\psi_0}{2} + \frac{\beta}{2} \text{tr}\left(\mathbf{K}_{\mathbf{uu}}^{-1}\boldsymbol{\psi}_2\right)\Bigg) - \text{KL}\left(q(\mathbf{X}|\boldsymbol{\theta}_V) \,\|\, p(\mathbf{X})\right) \Bigg] d\mathbf{Y} \\
&= \sum_{j=1}^{p} \left(-\frac{1}{2} \log \left|\mathbf{K}_{\mathbf{uu}} + \beta(\boldsymbol{\psi}_2 - {\boldsymbol{\psi}_1}^{\top}\boldsymbol{\psi}_1)\right| - \frac{1}{2} \log \left|\mathbf{K}_{\mathbf{uu}}^{-1}\right| - \frac{\beta\psi_0}{2} + \frac{\beta}{2} \text{tr}\left(\mathbf{K}_{\mathbf{uu}}^{-1}\boldsymbol{\psi}_2\right) \right) \\
&\qquad - \text{KL}\left(q(\mathbf{X}|\boldsymbol{\theta}_V) \,\|\, p(\mathbf{X})\right) + \underbrace{\log \int p(\boldsymbol{S}|\mathbf{Y}, \boldsymbol{\theta}_L) \prod_{j=1}^{p} \mathcal{N}\left(\mathbf{y}_{:,j}|\mathbf{0}, \hat{\mathbf{K}}_j\right) d\mathbf{Y}}_{\text{Laplace approximation}} . \qquad (5.27)
\end{aligned}
$$

In many cases, the likelihood is assumed to factorise across output dimensions, $p(\boldsymbol{S}|\mathbf{Y}, \boldsymbol{\theta}_L) = \prod_{j=1}^{p} p(\boldsymbol{s}_{:,j}|\mathbf{y}_{:,j})$, resulting in a series of p integrals,

$$
\log \int p(\boldsymbol{S}|\mathbf{Y}, \boldsymbol{\theta}_L) \prod_{j=1}^{p} \mathcal{N}\left(\mathbf{y}_{:,j}|\mathbf{0}, \hat{\mathbf{K}}_j\right) d\mathbf{Y} = \sum_{j=1}^{p} \log \int p(\boldsymbol{s}_{:,j}|\mathbf{y}_{:,j}) \mathcal{N}\left(\mathbf{y}_{:,j}|\mathbf{0}, \hat{\mathbf{K}}_j\right) d\mathbf{y}_{:,j}.
$$

The approximate bound $L(q_{\mathbf{X}})_{\text{lap}}$ then becomes completely factorised across dimensions, with the exception of the $\text{KL}\left(q(\mathbf{X}|\boldsymbol{\theta}_V) \,\|\, p(\mathbf{X})\right)$ term.

Real data is frequently collected from a variety of sources, and relating to some underlying latent system. For example in a health scenario, it may be that a patients underlying general health explains the data that is collected. The data collected may take the form of: survey answers where the data is ordinal; white blood cell counts that provide count data; the presence of genes producing binary data; and a multitude of other experiments that may give continuous data. From a generative modelling perspective, each of the above sources requires a different likelihood. Equation (5.27) shows that the bound factorises across output dimensions if the likelihood also factorises across output dimensions. It is then possible to assume a separate likelihood for each output dimension of $\boldsymbol{S}$. As previously mentioned, the MICE (Buuren and Groothuis-Oudshoorn, 1999) imputation method also provides this capability of assuming alternative noise models for each dimension; however the PCA, GPLVM, BGPLVM methods do not, they must assume Gaussian corruptions.

By using a separate likelihood for each output dimension, the integral in Equation (5.27) will differ for each of the p output dimensions. Using the results outlined in Section 3.2.1, and specifically the marginal likelihood following a Laplace approximation, Equation (3.11), a Laplace approximation can be made for each log marginal likelihood term,

$$\log \prod_{j=1}^{p} \int p(\boldsymbol{s}_{:,j}|\mathbf{y}_{:,j}) \mathcal{N}\left(\mathbf{y}_{:,j}|\mathbf{0}, \hat{\mathbf{K}}_j\right) d\mathbf{y}_{:,j} \tag{5.28}$$

$$= \sum_{j=1}^{p} \log p(\boldsymbol{s}_{:,j}|\hat{\mathbf{y}}_{:,j}) - \frac{1}{2}\hat{\mathbf{y}}_{:,j}^{\top}\hat{\mathbf{K}}_j^{-1}\hat{\mathbf{y}}_{:,j} - \frac{1}{2}\log\left|\mathbf{I} + \boldsymbol{W}_j^{\frac{1}{2}}\hat{\mathbf{K}}_j\boldsymbol{W}_j^{\frac{1}{2}}\right|, \tag{5.29}$$

where $\boldsymbol{W}_j = -\frac{d^2 \log p(\boldsymbol{s}_{:,j}|\hat{\mathbf{y}}_{:,j})}{d\hat{\mathbf{y}}_{:,j}^2}$ denotes the Hessian for the likelihood of the jth dimension, $p(\boldsymbol{s}_{:,j}|\mathbf{y}_{:,j})$, with the jth column of $\boldsymbol{s}_{:,j}$. $\hat{\mathbf{y}}_{:,j}$ denotes the mode of the posterior and is found through Newton's method, for the jth dimension. If the likelihood is separate for each output dimension; p Laplace approximations must be made to evaluate the approximate lower bound, $L(q_{\mathbf{X}})_{\text{lap}}$, increasing the computational burden similarly to the missing data scenario for the BGPLVM in Section 5.5.1; where the missingness pattern is different for each output. However it also allows each output dimension of Equation (5.27), to have a different noise variance, β_j^{-1} for each dimension j, at no extra computational cost. Allowing β_j^{-1} to vary amongst output dimensions can have some benefits, particularly when the likelihoods differ for each output.

The final form of the bound with a factorising likelihood is then,

$$\begin{aligned} L(q_{\mathbf{X}})_{\text{lap}} = \sum_{j=1}^{p} \Bigg(& -\frac{1}{2}\log\left|\mathbf{K}_{\mathbf{uu}} + \beta_j(\boldsymbol{\psi}_2 - \boldsymbol{\psi}_1{}^{\top}\boldsymbol{\psi}_1)\right| - \frac{1}{2}\log\left|\mathbf{K}_{\mathbf{uu}}^{-1}\right| - \frac{\beta_j\psi_0}{2} + \frac{\beta_j}{2}\text{tr}\left(\mathbf{K}_{\mathbf{uu}}^{-1}\boldsymbol{\psi}_2\right) \\ & + \log p(\boldsymbol{s}_{:,j}|\hat{\mathbf{y}}_{:,j}) - \frac{1}{2}\hat{\mathbf{y}}_{:,j}^{\top}\hat{\mathbf{K}}_j^{-1}\hat{\mathbf{y}}_{:,j} - \frac{1}{2}\log\left|\mathbf{I} + \boldsymbol{W}^{\frac{1}{2}}{}_j\hat{\mathbf{K}}_j\boldsymbol{W}^{\frac{1}{2}}{}_j\right|\Bigg) \\ & - \text{KL}\left(q(\mathbf{X}|\boldsymbol{\theta}_V)\,\|\,p(\mathbf{X})\right). \end{aligned} \tag{5.30}$$

To optimise the approximate variational lower bound, derivatives of $L(q_{\mathbf{X}})_{\text{lap}}$ must be taken with respect to the likelihood parameters, $\boldsymbol{\theta}_L$, β_j, and the matrices ψ_0, $\boldsymbol{\psi}_1$, $\boldsymbol{\psi}_2$. The chain rule can then be used to compute the derivatives with respect to the variational parameters of $q(\mathbf{X}|\boldsymbol{\theta}_V)$, and the kernel parameters, $\boldsymbol{\theta}_K$. Details key to obtaining these non-trivial derivatives can be found in the Appendix E.1.1.

5.6.4 Missing Data Bayesian Gaussian Process Latent Variable Model for Non-Gaussian Likelihoods

Missing data for the LABGPLVM model can be handled in an analogical way to the missing data BGPLVM model put forward in Section 5.5.1. The approximate variational bound of the LABGPLVM, Equation (5.27), can be generalised similarly to the BGPLVM missing data model, Equation (5.16) (although it is now the observed $\boldsymbol{S}$ that has missing values, not the now latent $\mathbf{Y}$). This provides an approximate variational lower bound on the marginal likelihood for the observed data $\log p(\boldsymbol{S}^{\mathcal{O}}|\boldsymbol{\theta}_L) \gtrsim L(q_{\mathbf{X}^{\mathcal{O}}})_{\text{lap}}$,

$$\begin{aligned} L(q_{\mathbf{X}^{\mathcal{O}}})_{\text{lap}} = \sum_{j=1}^{p} \Bigg(& -\frac{1}{2}\log\left|\mathbf{K}_{\mathbf{uu}} + \beta_j({\boldsymbol{\psi}_2}^{\mathcal{O}_j} - {\boldsymbol{\psi}_1}^{\mathcal{O}_j\top}{\boldsymbol{\psi}_1}^{\mathcal{O}_j})\right| - \frac{1}{2}\log\left|\mathbf{K}_{\mathbf{uu}}^{-1}\right| - \frac{\beta_j {\psi_0}^{\mathcal{O}_j}}{2} \\ & + \frac{\beta_j}{2}\text{tr}\left(\mathbf{K}_{\mathbf{uu}}^{-1}{\boldsymbol{\psi}_2}^{\mathcal{O}_j}\right)\Bigg) - \text{KL}\left(q(\mathbf{X}|\boldsymbol{\theta}_V)\,\|\,p(\mathbf{X})\right) \\ & + \log \underbrace{\int p(\boldsymbol{S}^{\mathcal{O}}|\mathbf{Y}^{\mathcal{O}}) \prod_{j=1}^{p} \mathcal{N}\left(\mathbf{y}_{:,j}^{\mathcal{O}_j}|\mathbf{0}, \hat{\mathbf{K}}^{\mathcal{O}_j}\right) d\mathbf{Y}^{\mathcal{O}}}_{\text{Laplace approximation}} . \end{aligned} \tag{5.31}$$

Analogically to Equation (5.29), if the likelihood factorises,

$$\begin{aligned} \log & \int p(\boldsymbol{S}^{\mathcal{O}}|\mathbf{Y}^{\mathcal{O}}) \prod_{j=1}^{p} \mathcal{N}\left(\mathbf{y}_{:,j}^{\mathcal{O}_j}|\mathbf{0}, \hat{\mathbf{K}}^{\mathcal{O}_j}\right) d\mathbf{Y}^{\mathcal{O}} \\ & = \sum_{j=1}^{p} \log p(\boldsymbol{s}_{:,j}^{\mathcal{O}_j}|\hat{\mathbf{y}}_{:,j}^{\mathcal{O}_j}) - \frac{1}{2}(\hat{\mathbf{y}}_{:,j}^{\mathcal{O}_j})^{\top}(\hat{\mathbf{K}}^{\mathcal{O}_j})^{-1}\hat{\mathbf{y}}_{:,j}^{\mathcal{O}_j} \\ & \qquad - \frac{1}{2}\log\left|\mathbf{I} + (\boldsymbol{W}^{\mathcal{O}_j})^{\frac{1}{2}}\hat{\mathbf{K}}^{\mathcal{O}_j}(\boldsymbol{W}^{\mathcal{O}_j})^{\frac{1}{2}}\right|, \end{aligned} \tag{5.32}$$

where in this case the Hessian, where $\boldsymbol{W}^{\mathcal{O}_j} = -\frac{d^2 \log p(\boldsymbol{s}_{:,j}^{\mathcal{O}_j}|\hat{\mathbf{y}}_{:,j}^{\mathcal{O}_j})}{d(\hat{\mathbf{y}}_{:,j}^{\mathcal{O}_j})^2}$, is only computed the data with observations $\boldsymbol{s}_{:,j}^{\mathcal{O}_j}$, for output j. $\hat{\mathbf{K}}^{\mathcal{O}_j}$ refers to n_j rows and columns of $\hat{\mathbf{K}}$ that correspond to the observed values of $\boldsymbol{s}_{:,j}^{\mathcal{O}_j}$

The approximate lower bound given in the this section and the bound for BGPLVM with missing data, Section 5.5.1, can be minimised with respect to the variational parameters of $q(\mathbf{X}|\boldsymbol{\theta}_V)$, namely the mean and variance as well as the kernel hyper-parameters and likelihood parameters. $q(\mathbf{X}|\boldsymbol{\theta}_V)$ then provides a lower dimensional representation of the partially missing observation data, $\mathbf{Y}$ or $\boldsymbol{S}$, as typically the dimensionality of the input $\mathbf{X}$ will be smaller than the number of output measurements.

For imputation, predictions must be made for the missing values, $\mathbf{Y}^{\mathcal{M}}$ or $\boldsymbol{S}^{\mathcal{M}}$, using the variational posterior learnt for the uncertain inputs, $q(\mathbf{X}|\boldsymbol{\theta}_V)$.

From the insights given in Section 5.6.2 it is clear that to make predictions for imputations, predictions must be first made for $p(\boldsymbol{F}|\mathbf{X}, \boldsymbol{\theta}_L, \boldsymbol{\theta}_K)$ and $p(\mathbf{Y}|\mathbf{X}, \boldsymbol{\theta}_L, \boldsymbol{\theta}_K)$ for the BGPLVM model, and LABGPLVM model respectively, in a similar fashion to a sparse Gaussian process. However in this case the input $\mathbf{X}$ is itself uncertain, and its approximate posterior, $q(\mathbf{X}|\boldsymbol{\theta}_V)$, has been learnt whilst taking into account the non-missing observations in $\mathbf{Y}$ and $\boldsymbol{S}$. Note that this differs from the complete case analysis discussed in Section 5.1, whereby only complete rows of $\mathbf{Y}$ and $\boldsymbol{S}$ would be considered to use for imputation. The predictions made for $\boldsymbol{F}$ or $\mathbf{Y}$ can then be used to sample from the appropriate likelihood for each output dimension. This sample provides an example imputed dataset, $\mathbf{Y}_m$, for the missing data BGPLVM model where the sample is taken from a Gaussian; or in the case of the LABGPLVM, $\boldsymbol{S}_m$, from the likelihood associated with that output.

The experiments that follow will use these sampled imputed datasets to evaluate the quality of the imputation, against known missing values, treated as test-data. They will also indicate where the BGPLVM and LABGPLVM missing data models are most appropriate for imputation in practice, in comparison with existing popular imputation techniques. We defer the discussion of how these predictive functions $p(\boldsymbol{F}|\mathbf{X}, \boldsymbol{\theta}_L, \boldsymbol{\theta}_K)$ and $p(\mathbf{Y}|\mathbf{X}, \boldsymbol{\theta}_L, \boldsymbol{\theta}_K)$ with uncertain inputs $q(\mathbf{X}|\boldsymbol{\theta}_V)$ are obtained to Chapter 6.

5.6.5 Experiments

In this section we describe a series of simulation studies that have been carried out to uncover limitations and applicability. The LABGPLVM builds upon the missing data BGPLVM model (Section 5.5.1) however a study of the BGPLVMs applicability and advantages in the context of imputation has not previously been carried out. The LABGPLVM additionally contains the original BGPLVM as special case. i.e. when the likelihood is a Gaussian distribution with extremely small variance, acting as a Kronecker delta function, $p(\boldsymbol{s}_{:,j}|\mathbf{y}_{:,j}) = \mathcal{N}\left(\boldsymbol{s}_{:,j}|\mathbf{y}_{:,j}, \beta_l^{-1}\mathbf{I}\right)$ at $\lim_{\beta_l^{-1}\to 0}$. This causes the approximate bound of the LABGPLVM to collapse to the bound of the BGPLVM, where $\mathbf{Y}$ is replaced with $\boldsymbol{S}$.

Simulation studies are carried out where the input data are sampled from a low dimensional space, $\mathbf{X} \sim \mathcal{N}(\mathbf{X}|\mathbf{0}, \mathbf{I}) \in \mathbb{R}^{n\times q}$, and are then projected through a series of p non-linear Gaussian process functions, into a higher dimensional space, $\boldsymbol{f}_{:,j} \sim \mathcal{N}(\mathbf{0}, k(\mathbf{X}, \mathbf{X}))$, where each output $\boldsymbol{f}_{:,j}$ shares the same input, $\mathbf{X}$. In these

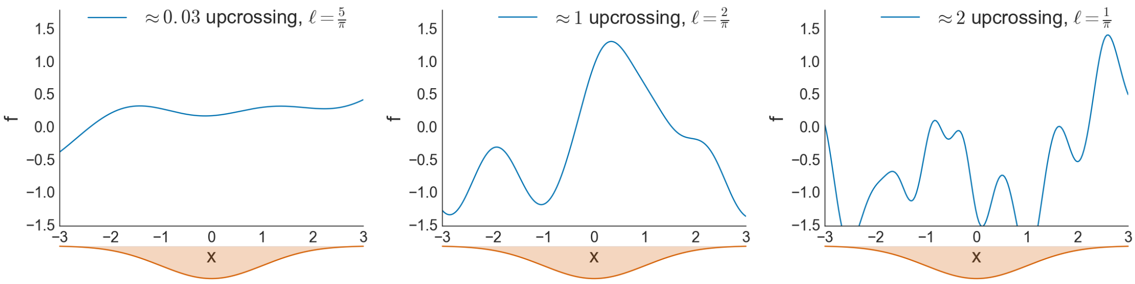


Fig. 5.2 Lengthscales chosen such that the expected number of upcrossings of $\boldsymbol{f}_{:,j}$ within span of the probable input $\mathbf{x}$ is known.

simulations a Gaussian process with a RBF kernel is used with ARD, Section 3.1.1, producing non-linear functions $\boldsymbol{f}_{:,j}$ for each output dimension j.

We consider the effect of moving from a short-lengthscale, producing very non-linear functions, to a larger lengthscale, essentially producing linear functions. The expected number of upcrossings of a unit area can be calculated for the RBF kernel (Rasmussen and Williams, 2006). The lengthscale, ℓ is varied between $\frac{1}{\pi}$, $\frac{2}{\pi}$ and $\frac{5}{\pi}$. These lengthscales correspond to 0.03, 1, and 2 expected number of up-crossings through zero of the Gaussian process function $\boldsymbol{f}_{:,j}$, given that the input range of most probable values for $\mathbf{X} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$, is around 4 standard deviations, as indicated by examples in Figure 5.2. This allows us to explore the effect on the performance of the models when the function is very non-linear, towards linear.

When the Gaussian process projection is linear, and the likelihood is Gaussian, the models assumptions are closely matched by those of PCA, Section 5.3. It makes sense in this case to compare against PCA, to see at which point the advantage of the non-linear BGPLVM model becomes negligible, or considering the additional complexities of optimising such a model, detrimental. Additionally, comparisons are made with the popular MICE imputation technique. MICE allows for a number of regression models between the observations columns to be specified, forming a series of chained equations that produce imputed sample datasets. Popular choices for regression models within MICE for continuous variables include predictive mean matching (PMM) and Bayesian linear regression (normed). Mean filling is additionally used as a baseline models for imputation. See Buuren and Groothuis-Oudshoorn (2011) for details of these methods and code used for imputation.

We additionally consider the effect of increasing or decreasing the signal-to-noise ratio, by holding the kernel variance fixed at 1, and changing the likelihood variance between 0.01 and 0.1. The performance is measured by the MAE of the imputed values, from the true values, that were discarded during model training. NLPD cannot be used as a comparison as MICE does not produce a predictive density, only a set of imputed datasets. The percentage of missing observations is increased between 10%, 20%, and 30%. 5-folds were used to provide an estimate of the variance of each technique. A medium number of output dimensions were used, $p = 40$, and the true input dimension used was $q = 2$, and number of data $n = 200$. For the BGPLVM and LABGPLVM methods, $m = 60$ inducing points were used throughout. The variational mean parameters of the variational distribution, $q(\mathbf{X}|\boldsymbol{\theta}_V)$, were initialised using PCA, and the variational variances are sampled uniformly between 0.0 and 0.1, exclusive. The inducing inputs, $\mathbf{Z} \in \mathbb{R}^{m \times q}$, are chosen as a subset of the PCA initialised variational means. Lengthscales are constrained to be positive, as are the likelihood and variance parameters. The RBF kernel hyper-parameters are assigned Gamma priors, $p(\ell), p(\sigma^2_{rbf}) \sim \mathcal{G}(2, 2)$, and their posterior distribution was approximately integrated over using *central composite design* (CCD) (Rue et al., 2009) rather than using the MAP solution for predictions. As shown in the next chapter the posterior distribution of the hyper-parameters can become much less well determined when the inputs are uncertain, and consequently predictions need to take account of this additional uncertainty.

In the first set of experiments the observations, $\boldsymbol{S}$, were produced by corrupting the latent function Gaussian noise, $\boldsymbol{s}_{:,j} \sim \mathcal{N}\left(\boldsymbol{f}_{:,j}, \beta^{-1}\mathbf{I}\right)$, that must be subsequently learnt by PCA and the BGPLVM. For this type of data, the BGPLVM and the new LABGPLVM can be made to be equivalent; by using Gaussian likelihood for the latter model and fixing likelihood variance to a small value. In these experiments the likelihood variance β^{-1} was shared between all output dimensions, p, for the BGPLVM.

Figure 5.3 and Table 5.1 show the results for this simulation study. The BGPLVM model, outperforms MICE relatively consistently in this setting. Figure 5.3 shows that an increase in performance is most pronounced when the lengthscale is $\frac{2}{\pi}$, corresponding to a relatively non-linear function. As predicted the performance of PCA closely matches that of the BGPLVM when the lengthscale is very long, $\frac{5}{\pi}$, and sometimes even slightly outperforms the BGPLVM. When there is a large amount of missing data, 30%, and the function is very non-linear, $\frac{1}{\pi}$, the BGPLVM does not provide any increase in performance. This is likely due to the model being incapable of inferring the short lengthscale of the process, and so the log-likelihood is higher when it is modelled as a linear function, with a larger likelihood variance, producing an equivalent model to that of PCA. Increasing the SNR to 0.1 makes the effect more pronounced, as would be expected in this situation. Additional figures for 15% data can be found in Appendix E.5.

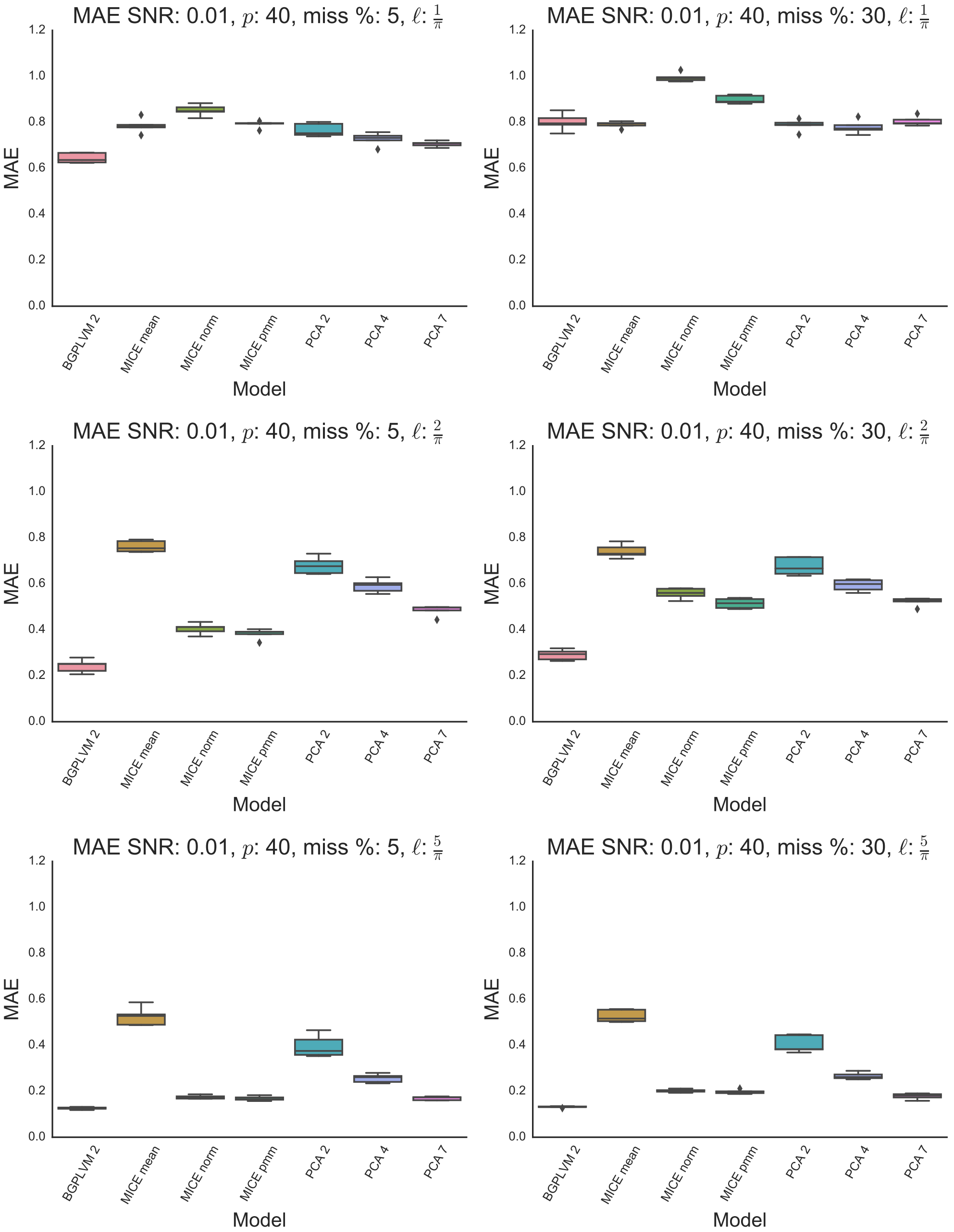


Fig. 5.3 Simulation study results for BGPLVM imputation model, with normally distributed observations. $\ell = \frac{1}{\pi}$ indicates a very non-linear function, $\ell = \frac{2}{\pi}$ indicates a relatively non-linear function, and $\ell = \frac{5}{\pi}$ indicates an almost linear function. Missingness levels are varied between 5% and 30%.

				BGPLVM	MICE	MICE	MICE	PCA	PCA	PCA
SNR	p	M%	ℓ	$q = 2$	mean	norm	pmm	$q = 2$	$q = 4$	$q = 7$
0.01	40	5	$\frac{1}{\pi}$	**0.64 ± .02**	0.78 ± .03	0.85 ± .02	0.79 ± .02	0.77 ± .03	0.73 ± .03	0.71 ± .01
			$\frac{2}{\pi}$	**0.24 ± .03**	0.76 ± .03	0.4 ± .02	0.38 ± .02	0.68 ± .04	0.59 ± .03	0.48 ± .02
			$\frac{5}{\pi}$	**0.13 ± .01**	0.53 ± .04	0.17 ± .01	0.17 ± .01	0.4 ± .05	0.26 ± .02	0.17 ± .01
		15	$\frac{1}{\pi}$	**0.72 ± .04**	0.78 ± .01	0.91 ± .01	0.85 ± .02	0.76 ± .03	0.75 ± .03	0.73 ± .02
			$\frac{2}{\pi}$	**0.25 ± .02**	0.73 ± .01	0.45 ± .02	0.42 ± .02	0.67 ± .02	0.59 ± .02	0.5 ± .02
			$\frac{5}{\pi}$	**0.13 ± .0**	0.52 ± .02	0.18 ± .01	0.18 ± .01	0.4 ± .03	0.26 ± .01	0.17 ± .02
		30	$\frac{1}{\pi}$	0.8 ± .04	0.79 ± .01	1.0 ± .02	0.9 ± .02	0.79 ± .03	**0.78 ± .03**	0.8 ± .02
			$\frac{2}{\pi}$	**0.29 ± .02**	0.74 ± .03	0.56 ± .02	0.51 ± .02	0.68 ± .04	0.59 ± .03	0.52 ± .02
			$\frac{5}{\pi}$	**0.13 ± .0**	0.53 ± .03	0.2 ± .01	0.2 ± .01	0.41 ± .04	0.27 ± .01	0.18 ± .01
0.1	40	5	$\frac{1}{\pi}$	**0.75 ± .02**	0.82 ± .04	0.97 ± .02	0.91 ± .01	0.81 ± .02	0.78 ± .03	0.76 ± .02
			$\frac{2}{\pi}$	**0.48 ± .02**	0.8 ± .02	0.67 ± .03	0.63 ± .02	0.73 ± .04	0.65 ± .02	0.56 ± .02
			$\frac{5}{\pi}$	0.38 ± .01	0.58 ± .05	0.47 ± .02	0.45 ± .02	0.47 ± .05	0.37 ± .02	**0.32 ± .01**
		15	$\frac{1}{\pi}$	**0.78 ± .01**	0.81 ± .01	1.03 ± .01	0.96 ± .01	0.8 ± .02	0.79 ± .03	0.79 ± .02
			$\frac{2}{\pi}$	**0.48 ± .02**	0.77 ± .01	0.71 ± .02	0.66 ± .01	0.72 ± .02	0.65 ± .02	0.59 ± .04
			$\frac{5}{\pi}$	0.39 ± .01	0.57 ± .02	0.49 ± .01	0.46 ± .01	0.47 ± .03	0.38 ± .01	**0.33 ± .01**
		30	$\frac{1}{\pi}$	0.85 ± .03	**0.82 ± .01**	1.1 ± .01	0.99 ± .01	0.83 ± .02	0.83 ± .02	0.87 ± .01
			$\frac{2}{\pi}$	**0.52 ± .01**	0.78 ± .02	0.78 ± .02	0.72 ± .02	0.72 ± .04	0.65 ± .02	0.61 ± .02
			$\frac{5}{\pi}$	0.4 ± .01	0.58 ± .03	0.52 ± .01	0.48 ± .01	0.48 ± .03	0.39 ± .01	**0.35 ± .01**

Table 5.1 Results comparing imputation methods on simulated data, with normally distributed observations. M% encodes the percentage of the full matrix that is missing during training, and is imputed

In the next set of simulation studies we aim to quantify the enhancement in performance when the observed data is non-Gaussian. In this case, instead of just corrupting $\boldsymbol{f}_{:,j}$ with Gaussian noise, for 50% of the outputs, the samples $\mathbf{y}_{:,j}$ are subsequently squashed through a probit function and sampled from a Bernoulli likelihood, $\pi_{:,j} = \Phi(\mathbf{y}_{:,j})$. The other 50% of outputs are simulated equivalently to the first set of experiments.

The LABGPLVM is assigned Bernoulli likelihoods for each of the output dimensions that contain binary observations, and Gaussian likelihoods for the dimensions that correspond to continuous observations. This illustrates two advantages of the LABGPLVM, it can not only handle a non-Gaussian likelihood, but a different likelihood may be used for each output dimension. The variances, β_j^{-1} of the latent function $\mathbf{y}_{:,j}$ for the Bernoulli outputs are allowed to vary independently, and a single separate likelihood variance for the Gaussian outputs is shared. Allowing the Bernoulli outputs to have a different variance provides the model with the capability of changing the slope of the probit transformation function. The standard BGPLVM model is trained as before, however an additonal MICE regression method is added that uses a Bernoulli noise model with a linear regressor, for each output corresponding to binary observations.

The results are shown in Figure 5.4 and Table 5.2, where Binary BGPLVM denotes the LABGPLVM with corresponding binary outputs. Similarly to in the first set of simulations, the results show that the increase in performance for the BGPLVM and the LABGPLVM are most pronounced when function is relatively non-linear, $\ell = \frac{2}{\pi}$. However the LABGPLVM performs poorly when $\frac{1}{\pi}$. When the SNR is increased to 0.1, the results become increasingly varied and all models perform similarly poorly, see Appendix E.5 for the associated box-plots; though again the LABGPLVM model remains competitive when the function is not extremely non-linear, $\ell = \frac{1}{\pi}$. When the lengthscale is short, or the noise ratio is high, then there is less correlation between different data-points. It seems that when the Gaussian process prior is expressing less dependence between the data-points, implying prior independence, such as when the noise variance is high, or the lengthscale very small, the approximation doesn't perform well. It is evident from the error bars and outliers associated with the predictions in this setting that the LABGPLVM has problems identifying the correct parameter settings. When the parameters are found correctly the model is competitive, otherwise it performs poorly. Note however that in this setting the model with the highest performance is the mean method; this simply takes the mean column as the prediction for every missing output. This indicates that this setting is particularly challenging for all models.

This challenge arises when the noise is high, i.e. SNR is 0.1, or the lengthscale is short; the sampled Bernoulli observations then become indistinguishable from noise so that the form of the producing function $\boldsymbol{f}_{:,j}$ becomes extremely difficult to learn. Bernoulli observations themselves are weakly informative, and so in the presence of so much noise, the observations appear almost completely random and the generative function is learnt to be completely flat with large variance.

These simulations provide an indication that the LABGPLVM provides some gains over the BGPLVM model when the observations are binary, and the signal to noise ratio is relatively low. The model however still has limitations; it does not perform when the function is very non-linear, or similarly when the function is very noisy. However, it is able to outperform MICE when the assumption that a lower dimensional representation of the data exists is accurate. This performance increase is maintained even when 30% of the data is missing.

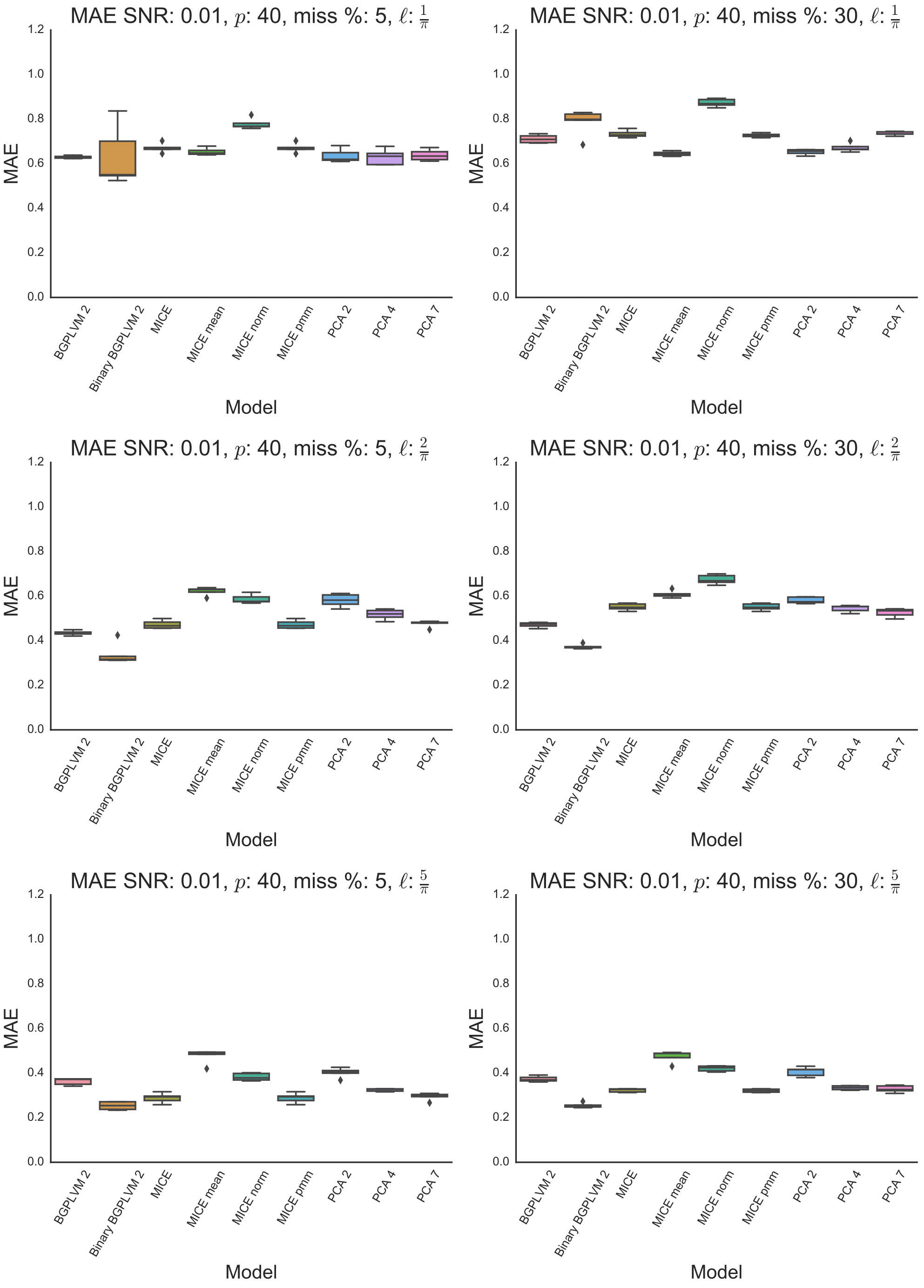


Fig. 5.4 Simulation study results for binary data imputation. $\ell = \frac{1}{\pi}$ indicates a very non-linear function, $\ell = \frac{2}{\pi}$ indicates a relatively non-linear function, and $\ell = \frac{5}{\pi}$ indicates an almost linear function. Missingness levels are varied between 5% and 30%.

				BGPLVM	BGPLVM	MICE	MICE	MICE	MICE	PCA	PCA	PCA
SNR	p	M%	ℓ		Binary	Mixed	mean	norm	pmm	$q = 2$	$q = 4$	$q = 7$
0.01	40	5	$\frac{1}{\pi}$	**0.63 ± .01**	0.63 ± .13	0.67 ± .02	0.65 ± .02	0.78 ± .02	0.67 ± .02	0.64 ± .03	0.63 ± .04	0.64 ± .03
			$\frac{2}{\pi}$	0.43 ± .01	**0.34 ± .05**	0.47 ± .02	0.62 ± .02	0.59 ± .02	0.47 ± .02	0.58 ± .03	0.52 ± .02	0.48 ± .01
			$\frac{5}{\pi}$	0.36 ± .02	**0.25 ± .02**	0.29 ± .02	0.48 ± .03	0.38 ± .02	0.29 ± .02	0.4 ± .02	0.32 ± .01	0.29 ± .02
		15	$\frac{1}{\pi}$	0.64 ± .02	0.7 ± .12	0.7 ± .02	0.65 ± .01	0.82 ± .02	0.7 ± .02	0.64 ± .01	**0.63 ± .01**	0.64 ± .02
			$\frac{2}{\pi}$	0.44 ± .01	**0.34 ± .03**	0.5 ± .03	0.61 ± .01	0.61 ± .03	0.5 ± .03	0.57 ± .02	0.52 ± .02	0.48 ± .02
			$\frac{5}{\pi}$	0.36 ± .01	**0.25 ± .02**	0.3 ± .01	0.47 ± .03	0.39 ± .01	0.3 ± .01	0.4 ± .02	0.32 ± .01	0.3 ± .02
		30	$\frac{1}{\pi}$	0.71 ± .02	0.79 ± .06	0.73 ± .02	**0.64 ± .01**	0.87 ± .02	0.73 ± .01	0.65 ± .01	0.67 ± .02	0.74 ± .01
			$\frac{2}{\pi}$	0.47 ± .01	**0.37 ± .01**	0.55 ± .02	0.61 ± .02	0.67 ± .02	0.55 ± .02	0.58 ± .01	0.54 ± .02	0.53 ± .02
			$\frac{5}{\pi}$	0.37 ± .01	**0.26 ± .01**	0.32 ± .01	0.47 ± .03	0.42 ± .01	0.32 ± .01	0.41 ± .02	0.33 ± .01	0.33 ± .02
0.10		5	$\frac{1}{\pi}$	0.68 ± .03	0.69 ± .09	0.72 ± .02	0.67 ± .02	0.83 ± .03	0.72 ± .02	0.66 ± .03	**0.65 ± .04**	0.68 ± .04
			$\frac{2}{\pi}$	0.51 ± .01	**0.46 ± .02**	0.57 ± .01	0.64 ± .02	0.68 ± .02	0.57 ± .01	0.61 ± .03	0.55 ± .02	0.53 ± .02
			$\frac{5}{\pi}$	0.45 ± .01	0.42 ± .03	0.43 ± .02	0.5 ± .03	0.53 ± .01	0.43 ± .02	0.44 ± .02	0.38 ± .01	**0.37 ± .01**
		15	$\frac{1}{\pi}$	0.71 ± .03	0.84 ± .01	0.73 ± .01	0.66 ± .01	0.86 ± .01	0.74 ± .02	0.66 ± .01	**0.65 ± .01**	0.68 ± .02
			$\frac{2}{\pi}$	0.52 ± .01	**0.47 ± .03**	0.59 ± .01	0.63 ± .01	0.7 ± .02	0.58 ± .01	0.59 ± .02	0.55 ± .02	0.53 ± .01
			$\frac{5}{\pi}$	0.45 ± .02	0.42 ± .05	0.44 ± .01	0.5 ± .02	0.54 ± .01	0.44 ± .01	0.44 ± .02	0.38 ± .01	**0.38 ± .02**
		30	$\frac{1}{\pi}$	0.77 ± .02	0.84 ± .01	0.78 ± .01	**0.66 ± .01**	0.92 ± .01	0.78 ± .01	0.67 ± .01	0.7 ± .02	0.78 ± .01
			$\frac{2}{\pi}$	0.56 ± .01	**0.5 ± .02**	0.62 ± .01	0.63 ± .01	0.76 ± .02	0.62 ± .01	0.6 ± .01	0.58 ± .02	0.58 ± .02
			$\frac{5}{\pi}$	0.46 ± .01	0.42 ± .04	0.46 ± .01	0.5 ± .02	0.57 ± .01	0.46 ± .01	0.45 ± .02	**0.4 ± .01**	0.41 ± .01

Table 5.2 Results comparing imputation methods on simulated data with binary observations on half the output dimensions. M% encodes the percentage of the full matrix that is missing during training, and is imputed. MICE mixed indicates the mice model with mixed output likelihoods

5.6.6 Related Work

A number of related models for handling non-Gaussian data within Gaussian process latent variable models have been put forward in the literature, though to our knowledge, no existing models have been applied to missing data imputation.

Andrade (2015) approached the problem of non-normality by using the expectation propagation, in a similar manner to the method proposed here. Unfortunately expectation propagation is known to be a relatively expensive approximate algorithm, and does not always converge on a solution. If missing data exists in each output dimension, or a separate likelihood is assumed for each output, the approximation must be made for each dimension separately; which would be impractically expensive with EP. Gal et al. (2015) proposed an inference method for the BGPLVM for categorical data. This method does not collapse $q(\mathbf{u}|\boldsymbol{\theta}_V)$, and uses Monte Carlo sampling to approximately compute problematic integrals. This allows them to obtain approximate gradients with respect to the marginal likelihood. We believe this approach to inference would be appropriate for the missing data case with small modifications, and could additionally be used for a variety of other likelihoods. Buettner et al. (2014) consider inference within the GPLVM, with censored observations. They do not consider the further Bayesian generalisation and so are not able to obtain a rich representation of the latent space, encoding the posterior uncertainty of the latent variables, $p(\mathbf{X}|\mathbf{Y})$, and so during imputation, this uncertainty cannot be propagated to the missing values imputed. Barrett and Coolen (2015) proposed a similar approach, where a GPLVM is used to jointly learn a lower dimensional representation of survival data, and covariate data. The hope is that this lower dimensional representation may contain some structure of interest, or be suitable to regress from itself. We will consider similar architectures to this in the following chapter.

5.7 Conclusions and Future Work

This chapter proposed a non-Gaussian BGPLVM model for missing data, the LABGPLVM, a new imputation method that allows Bayesian Gaussian process based dimensionality reduction to be performed, with non-Gaussian likelihoods. The model allows separate likelihoods on each of its output dimensions; allowing a joint latent space to be learnt for data that shares an underlying representation in a lower dimensional space, but has different output data-types. The model's capabilities are demonstrated on a dataset containing both binary outputs and Gaussian outputs, and its performance was found to be competitive with commonly used imputation methods on a simulated dataset. The same method could be used when a wider variety of data types exist, as is common in clinical studies. When a Gaussian likelihood is used, the model reverts back to the BGPLVM, a model that has previously been shown to be an effective tool for dimensionality reduction.

A Laplace approximation was used to compute an approximation to a problematic integral over the non-Gaussian likelihood distribution. Although the Laplace approximation is known to be effective for some likelihoods, the model could be modified to use a different approximation method. Another natural choice to handle the problematic integration, given the success of the chained Gaussian process model in Chapter 4, would be an additional variational approximation. However, one of the shortcomings of BGPLVM and the LABGPLVM extension, is that optimisation of the variational bound, with respect to the variational hyper-parameters and likelihood parameters can be problematic. This is amplified by the difficult initialisation of these models. We speculate that it may be that the addition of a wider variety of non-Gaussian likelihoods would cause the variational parameters of the model to become even more difficult to optimise following an additional variational approximation.

In this chapter we have used the model to perform imputation for missing data, however imputation is usually the first step in an analysis that contains missing data. The subsequent step is often to regress using the imputed values, and the uncertainty associated with it. A common approach is to apply a series of regression models to each imputed dataset. Chapter 6 will show that the same LABGPLVM can be used for this regression phase, where the input is now uncertain, as a result of imputation. This requires only a single model to be fit, that takes into account the distribution over the uncertainty naturally. We will also discuss alternative architectures that could be used in future work to provide additional modelling flexibility for clinical data analysis.

6

Combining the Paradigms, Future Work, and Conclusions

In this chapter we consider how the dimensionality reduction techniques used for of imputation of clinical data, Chapter 5, can be combined with survival analysis regression, the subject of Chapter 4. Chapter 4 discussed existing approaches survival analysis, and provided a novel regression model for the case where $y_{i,j}$ is a time-to-event outcome. It specifically focussed on how existing survival analysis regression techniques can be endowed with additional modelling flexibility by allowing the latent function of the covariates, $g(\mathbf{x}_{i,:})$, to be modelled with a Gaussian process. This requires the use of a non-Gaussian likelihood within a Gaussian process regression model.

Chapter 5 considered the importance of obtaining the uncertainty associated with imputations of missing values. However, it did not discuss in detail how this uncertainty can be subsequently used in a clinical analysis.

In practice, imputation is usually the first step in an analysis. In multiple imputation (MI), having imputed a number of possible datasets, the task is usually to regress using these imputed values to assess the impact of the covariates on some another variable of interest, or make predictions. It is important that the uncertainty associated with the imputed values is propagated through the regression model; otherwise the predictions made will be unrealistically optimistic. Unfortunately not all popular methods of imputation calculate the uncertainty associated with the imputed values (hot-deck, median, etc.); and so this uncertainty cannot be propagated; MI methods are a key exception.

The MI approach to handling the imputation uncertainty is with three steps. The first step is to *impute* multiple datasets. A separate regression model is then fit to each imputed dataset to *analyse* each dataset individually. These regression analysis are then *pooled*, typically using Rubin's rule (Rubin, 2004), when appropriate. A popular MI method is MICE, and has widespread use throughout statistics as already discussed in Chapter 5.

From a Bayesian point of view, the uncertainty associated with a particular value to be imputed, is a probability distribution. Ideally there is no need to impute multiple datasets and create a series of regressions, by sampling from this distribution, if the regression model can deal with the uncertainty directly. Regression models typically assume that the input values, $\mathbf{X}$, are certain. They must additionally be able to cope with a variety of different outcome types by allowing different likelihood functions, depending on the variable of interest. An example output variable of particular interest in this work, is a failure time outcome.

The chained survival analysis model, Section 4.3.1, introduced a flexible model for modelling this type of data. Unfortunately this model does not consider the case where $x_{i,j}$ is uncertain; it is instead assumed that the input is known and fixed; and so in its present state is not an appropriate regression model for this problem. In Chapter 5 the BGPLVM model was shown as an example of a sparse Gaussian process, where the input is uncertain. In the context of dimensionality reduction and missing data imputation, the input $\mathbf{X}$ must be learnt. Its relationship to sparse Gaussian processes was made particularly clear by the reformulation of the bound uncovered in Section 5.6.1. The BGPLVM can approximately propagate the input uncertainty through a sparse GP regression model, $y^*_{i,j} = g(x^*_{i,j}) + \epsilon_{i,j}$, where $x^*_{i,j} \sim \mathcal{N}\left(\mu^*, \beta^{-1*}\right)$, for some predicted mean μ^* and predicted variance, β^{-1*}. Unfortunately this model makes the assumption that the noise model, $\epsilon_{i,j}$, is Gaussian and so again is not an appropriate regression model for non-Gaussian outputs.

The LABGPLVM, a non-Gaussian likelihood extension of the BGPLVM based on the Laplace approximation of the variational bound, Section 5.6, provided a means for handling both a non-Gaussian likelihood and uncertain inputs. The model's capabilities have so far been considered in the context of dimensionality reduction; in this chapter we begin by discussing how predictions are made with this model. We will observe how the uncertainty in inputs affects the uncertainty of the posterior predictions. Propagation of input uncertainty is required to accurately reflect the true uncertainty in the predictions. This will be followed by a discussion of possible extensions and architectures that would allow further modelling flexibility, beyond that already considered in this thesis. Finally we will reflect on this work and summarise its key findings and contributions.

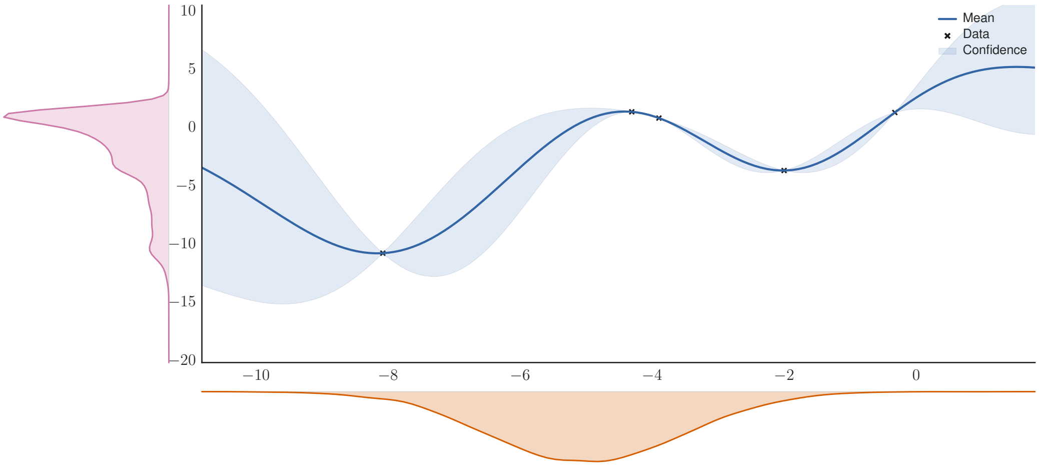


Fig. 6.1 The predictive distribution of a Gaussian process is not a Gaussian distribution if the mapping function is non-linear, and the input is Gaussian distribution

6.1 Input Uncertainty in Gaussian Processes

As discussed in Section 5.5, input uncertainty in Gaussian processes is problematic due to the integration of complex non-linear operator (matrix inverse) with respect to its inputs. The BGPLVM provides a manner in which a Gaussian approximation to the posterior of the inputs can be obtained, $p(\mathbf{X}|\mathbf{Y}) \approx q(\mathbf{X}|\boldsymbol{\theta}_V)$, by optimising a variational lower bound, Equation (5.23), even in the presence of missing output observations, Equation (5.16). Prediction with such a model however is non-trivial. By propagating a distribution through a non-linear Gaussian process, the resulting predictive distribution is no longer Gaussian; when a non-linear kernel function is used. The non-normality of the true predictive distribution is illustrated in Figure 6.1. When samples are taken from the Gaussian input, and propagated through the Gaussian process posterior, the resulting predictive distribution can be seen to be very non-Gaussian.

Uncertain input prediction has been studied by Girard et al. (2003), Candela et al. (2003) and Kuß (2006). Many subsequent publications have used these approaches for prediction when faced with uncertain inputs with a Gaussian process model (Damianou, 2015; Damianou et al., 2016; Deisenroth, 2010; Hensman and Lawrence, 2014). The approach proposed by Girard et al. (2003) makes a Gaussian approximation to the resulting non-Gaussian distribution, by recovering and matching its first two moments. It turns out that these moments can be calculated analytically (see Deisenroth (2010) for a thorough derivation).

The two problems that we aim to tackle for the remainder of this work are; predicting missing observations for the LABGPLVM imputation model in Section 5.6, and the implications of making predictions where the input uncertainty is provided, to propagate the imputed uncertainty. Uncertainty in the inputs for survival analysis could arise from either the preceding imputation, or known measurement error on the inputs during collection. For the former problem, the focus will be on the case where missing observations from an otherwise complete matrix, $\mathbf{Y}$, are required to be filled. This does not require predictions for new partially filled rows, $\mathbf{Y}^*$. We will consider the issues that exist for both the BGPLVM model, Section 5.5.1, and LABGPLVM, Section 5.6.4. Damianou and Lawrence (2015) additionally provides a method by which predictions can be made for new partially filled rows of $\mathbf{Y}^*$, that requires additional optimisation of an additional variational bound for the BGPLVM, to find the associated input uncertainty $q(\mathbf{X}^*|\boldsymbol{\theta}_V)$; a similar method can be used for the non-Gaussian likelihood case.

For predicting missing elements, $\mathbf{Y}^{\mathcal{M}}$, the bound that has been optimised for the missing data imputation, Section 5.5.1 and Section 5.6.4, provides an approximation to the input uncertainty, once observed elements have been taken into account $p(\mathbf{X}|\mathbf{Y}^{\mathcal{O}}) \approx q(\mathbf{X}|\boldsymbol{\theta}_V)$. Prediction then requires a sparse GP prediction at test points $\mathbf{X}^*$, with uncertainty in the input location, $q(\mathbf{X}^*|\boldsymbol{\theta}_V)$. Since $q(\mathbf{X}|\boldsymbol{\theta}_V)$ is factorised across datum, prediction is being made for missing elements in $\mathbf{Y}^{\mathcal{M}}$, and $\mathbf{X}^* \in \mathbf{X}$, then $q(\mathbf{X}^*|\boldsymbol{\theta}_V)$, are simply marginals of $q(\mathbf{X}|\boldsymbol{\theta}_V)$. The second problem, predicting with known measurement input uncertainty, also requires sparse GP prediction where uncertainty in the input location is integrated out. For notational convenience we will also call the distribution over the known input uncertainty, $q(\mathbf{X}^*|\boldsymbol{\theta}_V)$, but note that in this case no variational parameters must be learnt, they are explicitly provided. The prediction problem for this task remains the same as that of the BGPLVM, however for the LABGPLVM an additional corruption is assumed on the latent function.

Prediction for uncertain inputs requires the following integral for a sparse Gaussian process model, where inputs are uncertain,

$$p(\boldsymbol{F}^*|\mathbf{Y}) = \int p(\boldsymbol{F}^*|\mathbf{U}, \mathbf{Y}, \mathbf{X}^*) p(\mathbf{U}|\mathbf{Y}) p(\mathbf{X}^*|\mathbf{Y}) d\mathbf{U} d\mathbf{X}^*. \tag{6.1}$$

The variational bounds for sparse Gaussian processes, Section 3.3, and BGPLVM, Section 5.5, that have been optimised assume the following approximation,

$$p(\boldsymbol{F}^*|\mathbf{U}, \mathbf{Y}, \mathbf{X}^*) p(\mathbf{U}|\mathbf{Y}) \approx \prod_{j=1}^{p} p(\boldsymbol{f}^*_{:,j}|\mathbf{u}_{:,j}, \mathbf{X}^*) q(\mathbf{u}_{:,j}|\boldsymbol{\theta}_V)$$
$$p(\mathbf{X}^*|\mathbf{Y}) \approx q(\mathbf{X}^*|\boldsymbol{\theta}_V).$$

The integral then becomes,

$$p(\boldsymbol{F}^*|\mathbf{Y}) \approx \int \left[\int \prod_{j=1}^{p} p(\boldsymbol{f}^*_{:,j}|\mathbf{u}_{:,j}, \mathbf{X}^*) q(\mathbf{u}_{:,j}|\boldsymbol{\theta}_V) d\mathbf{u}_{:,j} \right] q(\mathbf{X}^*|\boldsymbol{\theta}_V) d\mathbf{X}^*. \tag{6.2}$$

Given an appropriate $q(\mathbf{u}_{:,j}|\boldsymbol{\theta}_V)$, the inner integral may be evaluated in closed form similarly to Equation (3.18),

$$\begin{aligned} p(\boldsymbol{F}^*|\mathbf{Y}) &\approx \int \prod_{j=1}^{p} q(\boldsymbol{f}^*_{:,j}|\boldsymbol{\theta}_V) q(\mathbf{X}^*|\boldsymbol{\theta}_V) d\mathbf{X}^* \\ q(\boldsymbol{f}^*_{:,j}|\boldsymbol{\theta}_V) &= \mathcal{N}\left(\boldsymbol{f}^*_{:,j}|\mathbf{K}_{\boldsymbol{f}^*\mathbf{u}}\mathbf{K}_{\mathbf{uu}}^{-1}\boldsymbol{\mu}_V, \mathbf{K}_{\boldsymbol{f}^*\boldsymbol{f}^*} - \mathbf{K}_{\boldsymbol{f}^*\mathbf{u}}\mathbf{K}_{\mathbf{uu}}^{-1}\mathbf{K}_{\mathbf{u}\boldsymbol{f}^*} + \mathbf{K}_{\boldsymbol{f}^*\mathbf{u}}\mathbf{K}_{\mathbf{uu}}^{-1}\Sigma_V\mathbf{K}_{\mathbf{uu}}^{-1}\mathbf{K}_{\mathbf{u}\boldsymbol{f}^*}\right). \end{aligned} \tag{6.3}$$

Note that $q(\boldsymbol{f}^*_{:,j}|\boldsymbol{\theta}_V)$ still depends on $\mathbf{X}^*$ through $\mathbf{K}_{\boldsymbol{f}^*\mathbf{u}}$, $\mathbf{K}_{\boldsymbol{f}^*\boldsymbol{f}^*}$ and $\mathbf{K}_{\mathbf{u}\boldsymbol{f}^*}$. Depending on the model, the variational mean parameter $\boldsymbol{\mu}_V$ and variational covariance parameter $\boldsymbol{\Sigma}_V$ of the variational distribution $q(\mathbf{u}_{:,j}|\boldsymbol{\theta}_V)$ differ in the following ways.

For a sparse-Gaussian processes, Section 3.3, trained with certain inputs, $q(\mathbf{u}_{:,j}|\boldsymbol{\theta}_V)$ is as in Equation (3.17), gives the optimal variational distributions parameters,

$$\begin{aligned} \boldsymbol{\mu}_V &\triangleq \beta\mathbf{K}_{\mathbf{uu}}(\mathbf{K}_{\mathbf{uu}} + \beta\mathbf{K}_{\mathbf{u}\boldsymbol{f}}\mathbf{K}_{\boldsymbol{f}\mathbf{u}})^{-1}\mathbf{K}_{\mathbf{u}\boldsymbol{f}}\mathbf{y}_{:,j} \\ \boldsymbol{\Sigma}_V &\triangleq \mathbf{K}_{\mathbf{uu}}(\mathbf{K}_{\mathbf{uu}} + \beta\mathbf{K}_{\mathbf{u}\boldsymbol{f}}\mathbf{K}_{\boldsymbol{f}\mathbf{u}})^{-1}\mathbf{K}_{\mathbf{uu}}, \end{aligned}$$

where $\boldsymbol{\mu}_V$ and $\boldsymbol{\Sigma}_V$ are the means and covariances of $q(\mathbf{u}|\boldsymbol{\theta}_V)$.

For the BGPLVM model, Section 5.5, where there is uncertainty on the inputs, $q(\mathbf{X}|\boldsymbol{\theta}_V)$, using Equations (5.14), (5.15), (see Equations (D.17) (D.18)) the mean and covariance become,

$$\boldsymbol{\mu}_V \triangleq \beta\mathbf{K}_{\mathbf{uu}}(\mathbf{K}_{\mathbf{uu}} + \beta\boldsymbol{\psi}_2)^{-1}\boldsymbol{\psi}_1{}^T\mathbf{y}_{:,j} \tag{6.4}$$

$$\boldsymbol{\Sigma}_V \triangleq \mathbf{K}_{\mathbf{uu}}(\mathbf{K}_{\mathbf{uu}} + \beta\boldsymbol{\psi}_2)^{-1}\mathbf{K}_{\mathbf{uu}}. \tag{6.5}$$

To propagate uncertainty in the case that we have a non-Gaussian output, as in the new LABGPLVM, we must derive the corresponding posterior distribution for our novel model, Section 5.6. The LABGPLVM also contains inducing inputs, $\mathbf{Z}$, directly corresponding to $\mathbf{U}$ as in the BGPLVM. Finding the approximate form of $p(\mathbf{u}_{:,j}|\boldsymbol{s}_{:,j})$ for this new model is more involved, and so the reader is referred to a verbose derivation in Appendix E.3. The resulting mean, $\boldsymbol{\mu}_V$ as in Equation (E.13), and covariance, $\boldsymbol{\Sigma}_V$

as in Equation (E.14), of the approximate posterior $q(\mathbf{u}|\boldsymbol{\theta}_V)$ are found to be,

$$\boldsymbol{\mu}_V = \mathbf{K}_{\mathbf{uu}}\boldsymbol{H}^{-1}{\boldsymbol{\psi}_1}^\top\beta_j\hat{\mathbf{y}}_{:,j} \tag{6.6}$$

$$\boldsymbol{\Sigma}_V = \mathbf{K}_{\mathbf{uu}}\left[\boldsymbol{H}^{-1} + \boldsymbol{H}^{-1}{\boldsymbol{\psi}_1}^\top\beta_j(\hat{\mathbf{K}}^{-1} + \boldsymbol{W}_j)^{-1}\beta_j\boldsymbol{\psi}_1\boldsymbol{H}^{-1}\right]\mathbf{K}_{\mathbf{uu}} \tag{6.7}$$

$$\boldsymbol{H} \triangleq (\mathbf{K}_{\mathbf{uu}} + \beta_j\boldsymbol{\psi}_2) \tag{6.8}$$

where to recap $\hat{\mathbf{y}}_{:,j}$ denotes the mode of the posterior, optimised through Newton's method, and $\boldsymbol{W}$ is the negative Hessian of the likelihood at this point.

Note that there is a posterior distribution for each output dimension j, in the standard BGPLVM these posterior distributions are the same when β^{-1} is shared amongst outputs. When missing data is considered, only the data that was observed for each output is used. Consequently the posterior distributions remain the same, however $\boldsymbol{\psi}_1$, $\boldsymbol{\psi}_2$, ψ_0, $\mathbf{y}_{:,j}$, $\hat{\mathbf{y}}_{:,j}$, $\hat{\mathbf{K}}$, and $\boldsymbol{W}_j$ are replaced by ${\boldsymbol{\psi}_1}^{\mathcal{O}_j}$, ${\boldsymbol{\psi}_2}^{\mathcal{O}_j}$, ${\psi_0}^{\mathcal{O}_j}$, $\mathbf{y}_{:,j}^{\mathcal{O}_j}$, $\hat{\mathbf{y}}_{:,j}^{\mathcal{O}_j}$, $\hat{\mathbf{K}}^{\mathcal{O}_j}$ and $\boldsymbol{W}^{\mathcal{O}_j}$ respectively. In this case predictions are made for each output independently, from a shared $q(\mathbf{X}|\boldsymbol{\theta}_V)$.

Unfortunately the outer integral of Equation (6.2) attempts to propagate a Gaussian distribution $q(\mathbf{X}|\boldsymbol{\theta}_V)$ through a Gaussian process. This results in a non-Gaussian distribution as discussed above, and is not analytically tractable. Damianou (2015) shows that using the work of Girard et al. (2003) for the BGPLVM the marginal moments for the each prediction location, when uncertain inputs integrated out, are given as follows,

$$\mathbb{E}_{q(\mathbf{X}^*|\boldsymbol{\theta}_V)}\left[\boldsymbol{f}^*\right] = {\boldsymbol{\mu}_V}^\top{\boldsymbol{\psi}_1}^*$$

$$\mathrm{Cov}_{q(\mathbf{X}^*|\boldsymbol{\theta}_V)}[\boldsymbol{f}^*] = {\boldsymbol{\mu}_V}^\top({\boldsymbol{\psi}_2}^* - {{\boldsymbol{\psi}_1}^*}^\top{\boldsymbol{\psi}_1}^*)\boldsymbol{\mu}_V + {\psi_0}^*\mathbf{I} - \mathrm{tr}\left((\mathbf{K}_{\mathbf{uu}}^{-1} - \mathbf{K}_{\mathbf{uu}}^{-1}\boldsymbol{\Sigma}_V\mathbf{K}_{\mathbf{uu}}^{-1}){\boldsymbol{\psi}_2}^*\right)\mathbf{I},$$

where

$${\psi_0}^* = \mathbb{E}_{q(\mathbf{X}^*|\boldsymbol{\theta}_V)}\left[\mathrm{tr}\left(\mathbf{K}_{\boldsymbol{f}^*\boldsymbol{f}^*}\right) \in \mathbb{R}^{1\times 1}\right]$$

$${\boldsymbol{\psi}_1}^* = \mathbb{E}_{q(\mathbf{X}^*|\boldsymbol{\theta}_V)}\left[\mathbf{K}_{\boldsymbol{f}^*\mathbf{u}}\right] \in \mathbb{R}^{1\times m}$$

$${\boldsymbol{\psi}_2}^* = \mathbb{E}_{q(\mathbf{X}^*|\boldsymbol{\theta}_V)}\left[\mathbf{K}_{\mathbf{u}\boldsymbol{f}^*}\mathbf{K}_{\boldsymbol{f}^*\mathbf{u}}\right] \in \mathbb{R}^{m\times m}.$$

These definitions for ψ_0, $\boldsymbol{\psi}_1$ and $\boldsymbol{\psi}_2$ are similar to the original Equations (5.13), (5.14), (5.15), but for test points, $\mathbf{X}^*$. In this case, $\boldsymbol{\mu}_V$ comes from the mean of the posterior covariance, Equation (6.4) and $\boldsymbol{\Sigma}_V$ comes from covariance of the posterior, Equation (6.5).

For the LABGPLVM, the posterior of $p(\mathbf{y}^*|\boldsymbol{S})$ is simply a Gaussian corruption of $p(\boldsymbol{f}^*|\boldsymbol{S})$. Once the appropriate $\boldsymbol{\Sigma}_V$ and $\boldsymbol{\mu}_V$ have been found the moments are then as follows,

$$\begin{aligned}\mathbb{E}_{q(\mathbf{X}^*|\boldsymbol{\theta}_V)}\left[\mathbf{y}^*\right] &= {\boldsymbol{\mu}_V}^\top {\boldsymbol{\psi}_1}^* \qquad (6.9)\\ \mathrm{Cov}_{q(\mathbf{X}^*|\boldsymbol{\theta}_V)}[\mathbf{y}^*] &= {\boldsymbol{\mu}_V}^\top ({\boldsymbol{\psi}_2}^* - {{\boldsymbol{\psi}_1}^*}^\top {\boldsymbol{\psi}_1}^*)\boldsymbol{\mu}_V + {\psi_0}^*\mathbf{I} - \mathrm{tr}\left((\mathbf{K}_{\mathbf{uu}}^{-1} - \mathbf{K}_{\mathbf{uu}}^{-1}\boldsymbol{\Sigma}_V\mathbf{K}_{\mathbf{uu}}^{-1}){\boldsymbol{\psi}_2}^*\right)\mathbf{I}\\ &\quad + \beta^{-1}\mathbf{I}, \qquad (6.10)\end{aligned}$$

where $\boldsymbol{\Sigma}_V$ is defined in Equation (6.7), and $\boldsymbol{\mu}_V$ is defined in Equation (6.6) for the LABGPLVM. A in-depth derivation of the above two results can be found in Appendix E.4 for the interested reader.

For the LABGPLVM, the Gaussian approximation to the posterior of the sparse Gaussian processes, $p(\mathbf{y}^*_{:,j}|\boldsymbol{S})$, can then be used to sample predicted values for the test observations, $\boldsymbol{S}^*$. These sampled values are used to compute multiple imputations of possible datasets in the experiments of Section 5.6.5.

In both the BGPLVM and the LABGPLVM for missing data, the posterior distribution of the inputs $q(\mathbf{X}|\boldsymbol{\theta}_V)$ is assumed to be factorised Gaussian, and the sparse Gaussian process predictive distribution of the latent function, $\boldsymbol{F}^*$, and $\mathbf{Y}^*$ respectively can be approximated by a Gaussian using moment matching as shown above. Since predictions can be made with a Gaussian input, the predictive distributions could be used as an input to a subsequent regression model. We show the effect of propagating input uncertainty through a regression model that has non-Gaussian outputs in the following section.

It should be noted, that when non-Gaussian outputs are imputed the sampled imputations then become non-Gaussian, and so $q(\mathbf{X}|\boldsymbol{\theta}_V)$ in the subsequent regression model would be non-Gaussian. In the following section we only consider normally distributed input uncertainty, and so leave this extension to future work.

6.2 Uncertain Input with Survival Models

As a demonstration of uncertainty propagation, we now use the LABGPLVM, Section 5.6, to perform non-linear Gaussian process survival analysis, with the log-logistic likelihood as in, Section 4.1.2; with censored failure time data, as well as uncertainty on the input. Problems such as these can arise in a medical setting, when there is measurement error on the input, or as shown in Chapter 5, as a result of imputing missing values.

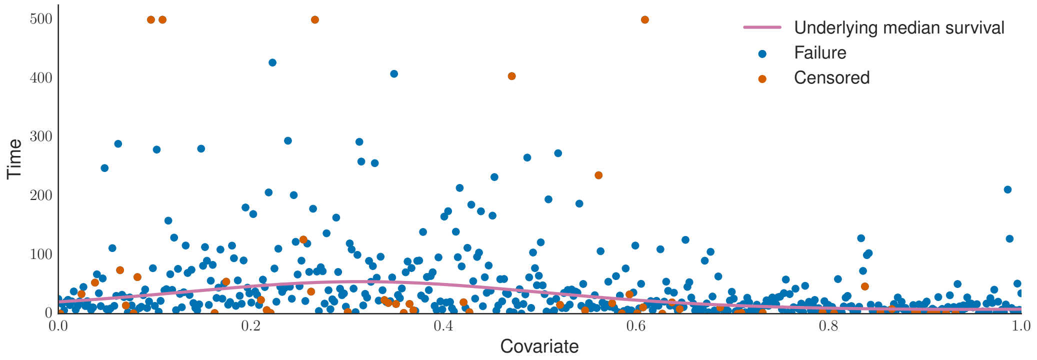
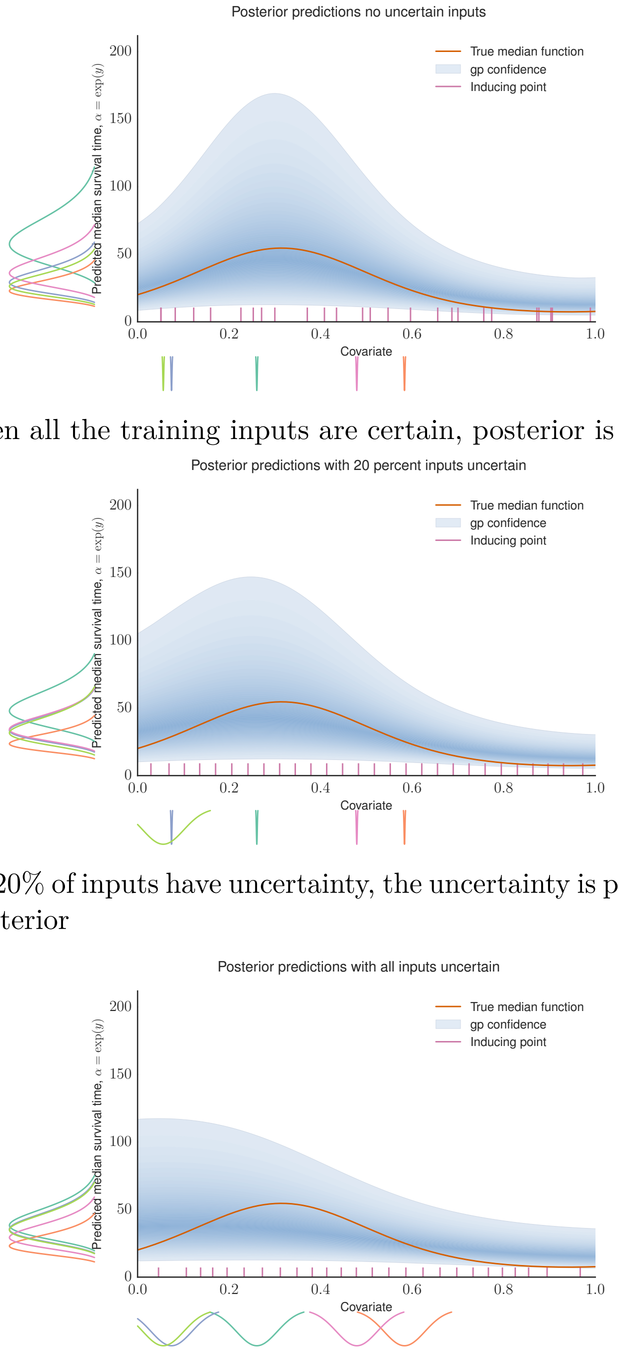


Fig. 6.2 Synthetic survival data

A synthetic dataset is used to illustrate how uncertainty on the input measurement, is propagated through to the regressions made on the predicted median failure time. Figure 6.2 shows the dataset that was used for training. It consists of 500 data, where a non-linear function of the covariates for the median failure time is used, $y_i = \exp(\sin(5x_i))$, shown in pink. The observed individual failure times $s_i = t_i$ are sampled from a log-logistic distribution, Section 4.1.2, with shape parameter, $\beta = 1.5$, and the scale parameter described by the median failure time function above, y_i. Random censoring is applied to 10% of the data, and all data $t_i > 500$ are also censored, simulating loss to follow up.

Three models were trained where each model is provided with a varying percentage of uncertainty in their inputs, $\mathbf{x}$. By introducing uncertainty in the input, it is hoped that the uncertainty is propagated through the model, and the posterior estimate of the median failure time function will be affected accordingly. Note that in this case, the true uncertainty of the input is 0, and so we expect the model with no uncertainty to have the highest accuracy.

Figure 6.3 shows the resulting inferences made when the model is provided with three different assumptions about its input $\mathbf{x}$. In Figure 6.3a, the model is provided with the true training data, such that all the uncertainty around all the values of $\mathbf{x}$ are essentially zero. The uncertainty associated with a random subset of 5 training input data-points is shown below the x-axis, and the resulting predictions for these inputs are shown to the left of the y-axis. As discussed in Section 5.6.1, by reducing the uncertainty of $q(\mathbf{X}|\boldsymbol{\theta}_V)$ to zero, the BGPLVM bound, that its non-Gaussian likelihood counterpart — the LABGPLVM — uses as a prior, falls back to a sparse Gaussian process model. In this case the model acts as a sparse Gaussian process AFT model similar to those discussed in Section 4.1.2, though unlike the chained survival analysis



(a) When all the training inputs are certain, posterior is narrow

(b) When 20% of inputs have uncertainty, the uncertainty is propagated to the posterior

(c) When all of inputs have uncertainty, the uncertainty in the posterior is increased

Fig. 6.3 Uncertain input regression with survival likelihood. By increasing the amount of uncertainty in the inputs provided for training, this figure shows how this uncertainty is propagated through to the posterior beliefs in the function. The true median failure time function, used for data generation, is marked in orange. CCD is used to approximately integrate over the uncertainty of the hyper-parameters, shown in Figure 6.4. The distributions below each figure show a random set of example training inputs, and the distributions on the left show the predictive distribution for these inputs once propagated through the posterior distribution, and subsequently approximated as in Section 6.1

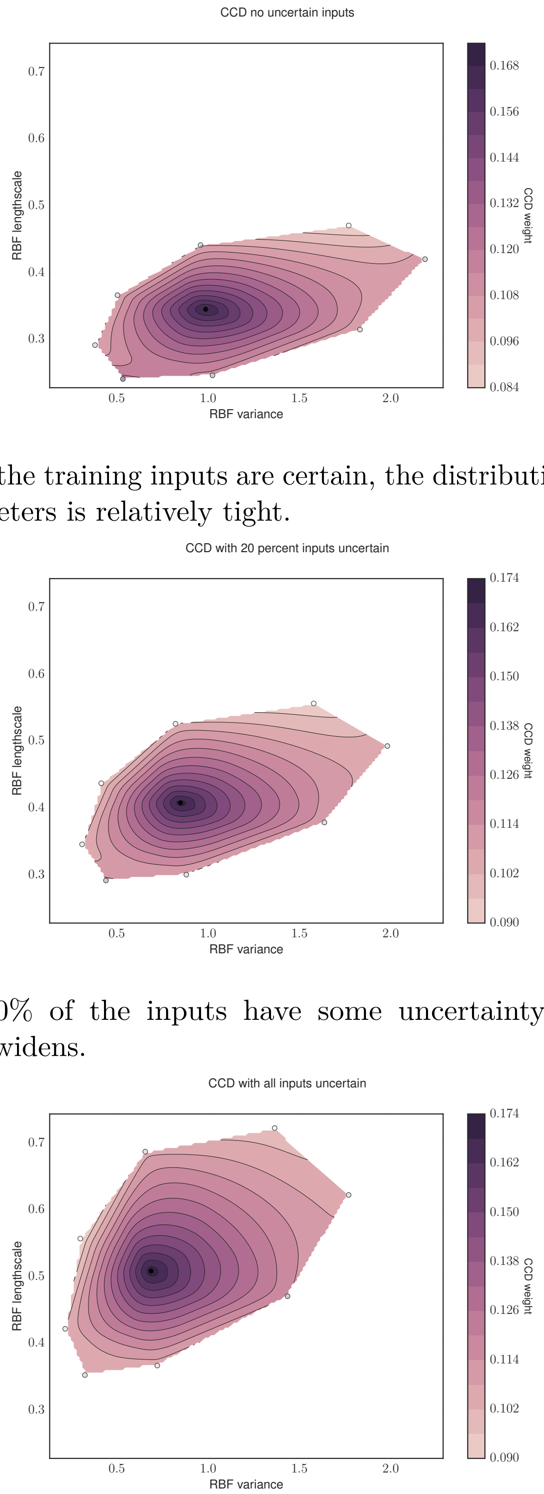
model, the shape is assumed to be fixed. The model fits the underlying function well as the true uncertainty is zero, the mean of the posterior passes directly through the true function, and the uncertainty is accurately reflected.

In Figure 6.3b the model is provided with 80% of its inputs known, however 20% are provided as a distribution over their true values, $\mathcal{N}\left(\mathbf{x}|\mathbf{x}_{\text{true}}, 0.002\right)$. As shown in Section 6.1, by predicting with an uncertain input, the resulting predictions may be non-Gaussian; however it is possible to find the moments of this non-Gaussian distribution, as is done in Figure 6.3a, note that they appear skewed as the logarithm of the median failure time is modelled with a Gaussian process, and so the predictions are shown in true space. It is clear that by increasing the uncertainty in the input, the lengthscale parameter becomes less well determined. This indicates that since the model is aware that the inputs may come from one of several input locations, it is probable that the outputs are also in practice spread across this input range, and so flatter functions are more plausible.

Finally Figure 6.3c shows a model in which all its training inputs are provided as uncertain. This further increases the belief that the function is of a longer lengthscale.

In practice to adjust the input uncertainty, the mean of $q(\mathbf{X}|\boldsymbol{\theta}_V)$, as in Equation (5.27) is set to the true value of $\mathbf{x}$, and the variance for the inputs that are uncertain during training, are set accordingly.

In Figure 6.3 the posterior uncertainty associated with the RBF kernel lengthscale and variance is approximately integrated over using *central composite design* (CCD) (Rue et al., 2009). In Gaussian processes CCD has been shown to be an effective replacement for MCMC sampling, while being significantly less computationally expensive (Vanhatalo et al., 2010, 2013). When uncertainty in the input is introduced, the mode of the posterior for these hyper-parameters becomes increasingly less peaked; and so a MAP approximation becomes less valid. In practice integration over the uncertainty of these hyper-parameters is an important step in faithfully representing the resulting uncertainty in predictions. Figure 6.4 shows the approximate posterior distribution of the two RBF kernel hyper-parameters that are integrated over, for each of the varying amount of training input uncertainty as in Figure 6.3. A Gamma prior, is used for each of the hyper-parameters, encouraging both parameters to stay relatively small, $p(\ell), p(\sigma^2_{rbf}) \sim \mathcal{G}(2, 2)$. By increasing the amount of uncertainty in the input, it becomes more probable that long lengthscale functions for the median failure time produced the data, since the inputs have the potential to be more spread across the input range; as was evident from the predictions that also used CCD. The difference is particularly apparent by comparing the posterior distribution of the hyper-parameters

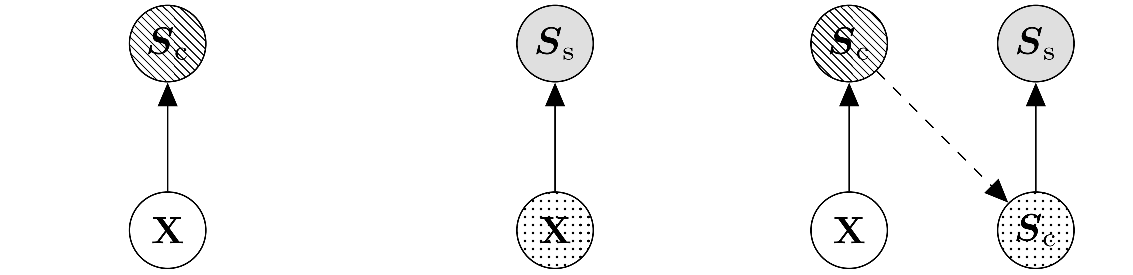


(a) When all the training inputs are certain, the distribution of the kernel hyper-parameters is relatively tight.

(b) When 20% of the inputs have some uncertainty, the posterior distribution widens.

(c) When all the inputs have uncertainty, the posterior distribution is wider still, shifting mass to longer lengthscale solutions.

Fig. 6.4 Estimated posterior distribution of the hyper-parameters RBF kernel for increasing number of uncertain inputs of the survival model, using CCD as an approximation



(a) BGPLVM/GPLVM imputation (b) Uncertain input non-Gaussian survival regression (c) Latent variable, followed by uncertain survival prediction

Fig. 6.5 Graphical models for combining imputation and survival regression

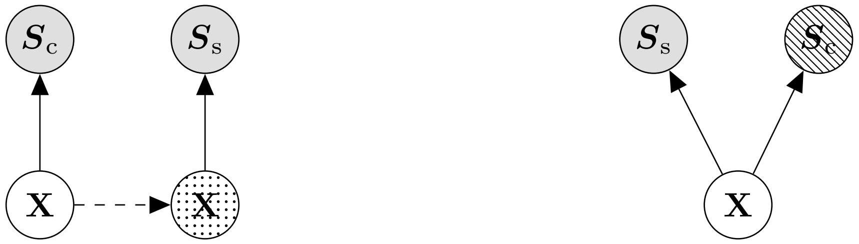
found when no input uncertainty is provided, Figure 6.4a, to the posterior found when all inputs are uncertain, Figure 6.4c.

6.3 Future Work

This thesis has considered the imputation of missing covariate data with latent variable models, and flexible Gaussian process based survival analysis regression methods, both with certain and uncertain inputs. The obvious next step in this direction of work would be to combine these two paradigms of imputation and regression. We will now consider how the novel works covered in this thesis could be used, and propose a variety of architectures that may form a basis of future research in this direction.

Graphical models of proposed architectures are given in Figures 6.5 and 6.6. Let $\boldsymbol{S}_\mathrm{c}$ denote covariates, and $\boldsymbol{S}_\mathrm{s}$ denote survival times. Empty nodes denote latent variables and filled nodes denote observed variables as usual. Striped nodes indicate partially missing observations, and dotted nodes indicate uncertain inputs. With this notation GPLVM, BGPLVM and LABGPLVM missing data models are shown in, Figure 6.5a, where a suitable likelihood is used for each output $\boldsymbol{s}_{:,j}$ of $\boldsymbol{S}_\mathrm{c}$. We ignore variational parameters and unobserved latent functions for notational simplicity in the graphical models; this allows the bigger picture to be seen without considering the inference algorithms required.

Uncertain input survival analysis regression as in the previous section is shown in Figure 6.5b. In future work it would be convenient to find a way to allow the input uncertainty, $q(\mathbf{X}|\boldsymbol{\theta}_V)$, to be non-Gaussian during training and prediction. One possible method would be to consider quadrature in order to integrate over the input uncertainty, as $q(\mathbf{X}|\boldsymbol{\theta}_V)$ usually is assumed to be a factorised mean field approximation.



(a) Learnt posterior of latent variables used to predict of survival

(b) Joint learn imputation and regression model

Fig. 6.6 Graphical models of architectures for imputation in future work

As alluded to, the next logical step for this line of work is to combine the latent variable imputation methods with survival analysis. This would require first imputing missing values with uncertainty, and then propagating the uncertainty through to the regression of survival, Figure 6.5c. If the BGPLVM model for missing data model was used for imputation, Section 5.5.1, then the LABGPLVM can be used for the regression model, as shown in the previous section, and so all the elements of this model are available.

Latent variable models aim to capture the underlying lower dimensional structure of the covariates, whilst removing the noise introduced by the likelihood. It may be that regression from the latent variables themselves to the survival times would be more effective, Figure 6.6a.

Alternatively, the survival times themselves may be able to explain some of the variation in the underlying latent space. In this case learning a joint model of the latent space would be appropriate, Figure 6.6b. Barrett and Coolen (2015) have provided preliminary results using the GPLVM model that suggest that this might be a promising direction of research. However by using the GPLVM rather than the BGPLVM they are not able to accurately capture the uncertainty associated with their optimised inputs. The new missing data LABGPLVM imputation model proposed in Section 5.6, could handle this joint model in the presence of missing covariate data, by using a log-logistic likelihood on one of its output dimensions to model the failure time data; similar to the Bernoulli observations used in the experiments, Section 5.6.5. In practice however survival data provides relatively weak information, particularly when a large amount of censoring occurs, and so we hypothesise this approach would provide only small gains in the presence of extreme censoring.

In addition to the above possible future directions, we hope to expand on the work carried out in Chapter 4 in a number of ways, particularly in regards to applications. Firstly we hope to demonstrate the applicability of the scalable chained survival analysis model on large scale datasets as these become available to the public domain. Secondly we hope to apply the beta chained Gaussian process model to a range of epidemiological tasks as previously proposed.

6.4 Conclusion

Throughout this body of research we have considered how the flexible non-parametric class of Gaussian process models can be used for issues relating to clinical data, primarily survival analysis regression, and clinical data imputation. Limitations with popular approaches have been relaxed, though the additional flexibility can also present its own problems; relatively complex inference methods often have to be used to provide approximations. These inference methods can be challenging to work with, requiring complex initialisation and optimisation, however the additional flexibility and in some cases scalability provided can be essential for accurately portraying uncertainty.

A number of contributions have been provided throughout this body of research, as well as providing a review of the key ideas that make clear how survival analysis regression methods can be combined with Gaussian process methods, focussing on the assumptions traditionally made by generalised linear models.

6.4.1 Summary of Contributions

- In Chapter 4 we proposed a new a flexible and scalable method for survival analysis regression, called the chained survival analysis model. It also showed that the same method can be used for an array of other modelling assumptions.

- In Chapter 5 we carried out the first set of experiments for a BGPLVM model for missing data in the context of imputation, showing promising results compared with a number of other imputation methods.

- Chapter 5 also contributed a second novel model, a non-Gaussian likelihood extension to the BGPLVM called the LABGPLVM, that allows a multitude of non-normally distributed observations to be handled.

- In this chapter, we discussed how the LABGPLVM can be applied to the context of uncertain input regression, with non-Gaussian likelihoods, including those used in survival analysis.

- This chapter also provided a number of future directions that would go some way to integrating the proposed clinical data imputation methods and novel uncertain input survival analysis regression method.

Appendix A

Useful identities

To follow the derivations of the key aspects of this thesis a number of identities will prove useful. The most significant and commonly used identities are a set of core Gaussian identities and matrix identities; these are reproduced here to aid the reader in following the derivations of this text. Many of the key identities can be found in many textbooks, see Bishop (2006) and Rasmussen and Williams (2006) for a more complete set.

A.1 Properties of Gaussian Distribution

A.1.1 Marginalization

The marginalization property of a joint Gaussian distribution allows for trivial marginalization of one of the variables. Given a joint Gaussian distribution $\mathcal{N}\left(\mathbf{x}|\boldsymbol{\mu},\boldsymbol{\Sigma}\right)$,

$$\mathbf{x} = \begin{bmatrix} \mathbf{x}_a \\ \mathbf{x}_b \end{bmatrix}, \boldsymbol{\mu} = \begin{bmatrix} \boldsymbol{\mu}_a \\ \boldsymbol{\mu}_b \end{bmatrix}, \qquad \boldsymbol{\Sigma} = \begin{bmatrix} \boldsymbol{\Sigma}_{aa} & \boldsymbol{\Sigma}_{ab} \\ \boldsymbol{\Sigma}_{ba} & \boldsymbol{\Sigma}_{bb} \end{bmatrix}$$

Then the marginal distributions are simply,

$$p(\mathbf{x}_a) = \mathcal{N}\left(\mathbf{x}_a|\boldsymbol{\mu}_a,\boldsymbol{\Sigma}_{aa}\right), \qquad p(\mathbf{x}_b) = \mathcal{N}\left(\mathbf{x}_b|\boldsymbol{\mu}_b,\boldsymbol{\Sigma}_{bb}\right)$$

Intuitively this indicates that if there are millions of variables that are jointly normally distributed, but the interest is in just one of those variables, then there is no need to integrate computationally across all the variables of no interest, in order to get the marginal distribution of the variable of interest. Most other distributions do not have this property, especially beyond bivariate distributions.

Additionally if one random variable, $\mathbf{y}$, is conditioned on another, $\mathbf{x}$, and the mean of the conditional distribution is a linear function of $\mathbf{x}$, marginalising with a Gaussian is simply,

$$p(\mathbf{x}) = \mathcal{N}\left(\mathbf{x}|\boldsymbol{\mu}, \boldsymbol{\Sigma}\right) \tag{A.1}$$

$$p(\mathbf{y}|\mathbf{x}) = \mathcal{N}\left(\mathbf{y}|\mathbf{A}\mathbf{x} + \mathbf{b}, \mathbf{L}^{-1}\right) \tag{A.2}$$

Then:

$$p(\mathbf{y}) = \mathcal{N}\left(\mathbf{y}|\mathbf{A}\boldsymbol{\mu} + \mathbf{b}, \mathbf{L}^{-1} + \mathbf{A}\boldsymbol{\Sigma}\mathbf{A}^{\top}\right) \tag{A.3}$$

$$\tag{A.4}$$

A.1.2 Conditioning

For Gaussian distributions the conditional distributions are also analytical,

$$p(\mathbf{x}) = \mathcal{N}\left(\mathbf{x}|\boldsymbol{\mu}, \boldsymbol{\Sigma}\right) \tag{A.5}$$

$$p(\mathbf{x}_a|\mathbf{x}_b) = \mathcal{N}\left(\mathbf{x}|\boldsymbol{\mu}_{a|b}, \boldsymbol{\Lambda}_{aa}^{-1}\right) \tag{A.6}$$

$$\boldsymbol{\mu}_{a|b} = \boldsymbol{\mu}_a - \boldsymbol{\Lambda}_{aa}^{-1}\boldsymbol{\Lambda}_{ab}(\mathbf{x}_b - \boldsymbol{\mu}). \tag{A.7}$$

For a conditonal distribution, and a prior, using the above definitions reversing the conditional is then,

$$p(\mathbf{x}) = \mathcal{N}\left(\mathbf{x}|\boldsymbol{\mu}, \boldsymbol{\Sigma}\right) \tag{A.8}$$

$$p(\mathbf{y}|\mathbf{x}) = \mathcal{N}\left(\mathbf{y}|\mathbf{A}\mathbf{x} + \mathbf{b}, \mathbf{L}^{-1}\right) \tag{A.9}$$

Then

$$p(\mathbf{x}|\mathbf{y}) = \mathcal{N}\left(\mathbf{x}|\mathbf{G}\{\mathbf{A}^{\top}\mathbf{L}(\mathbf{y} - \mathbf{b}) + \boldsymbol{\Sigma}^{-1}\boldsymbol{\mu}\}, \mathbf{G}\right) \tag{A.10}$$

$$\mathbf{G} = \left(\boldsymbol{\Sigma}^{\mathbf{-1}} + \mathbf{A}^{\top}\mathbf{L}\mathbf{A}\right)^{-1}. \tag{A.11}$$

A.2 Matrix Identities

Large parts of this research require strong familiarity with linear algebra. Two identities that are used on multiple occasions are the matrix inversion lemma, and the matrix determinant lemma.

A.2.1 Matrix Inversion Lemma

The Woodbury matrix identity, or *matrix inversion lemma* is a linear algebra identity as follows:

$$(A + UCV)^{-1} = A^{-1} - A^{-1}U(C^{-1} + VA^{-1}U)^{-1}VA^{-1} \tag{A.12}$$

It is often useful if inverting the matrix A is either computationally less expensive, or more numerically stable, than inverting the matrix $(A + UCV)^{-1}$. Conversely it is sometimes used in reverse, especially when inverting A or $(C^{-1} + VA^{-1}U)$ is numerically unstable, and if A is a positive diagonal matrix.

A.2.2 Matrix Determinant Lemma

Analogically to the matrix inversion lemma, the matrix determinant lemma provides a similar identity for matrix determinants

$$\left|A + UWV^T\right| = \left|W^{-1} + V^TA^{-1}U\right| |W| \, |A| \tag{A.13}$$

A.3 Expectation and Covariance

For a Gaussian the following equalities will prove useful,

$$\begin{aligned}
\mathrm{cov}[\boldsymbol{f}, \boldsymbol{f}^\top] &= \mathbb{E}\Big[(\boldsymbol{f} - \mathbb{E}\Big[\boldsymbol{f}\Big])(\boldsymbol{f}^\top - \mathbb{E}\Big[\boldsymbol{f}^\top\Big])\Big] \\
&= \mathbb{E}\Big[\boldsymbol{f}\boldsymbol{f}^\top - \boldsymbol{f}\mathbb{E}\Big[\boldsymbol{f}^\top\Big] - \mathbb{E}\Big[\boldsymbol{f}\Big]\boldsymbol{f}^\top + \mathbb{E}\Big[\boldsymbol{f}\Big]\mathbb{E}\Big[\boldsymbol{f}^\top\Big]\Big] \\
&= \mathbb{E}\Big[\boldsymbol{f}\boldsymbol{f}^\top\Big] - \mathbb{E}\Big[\boldsymbol{f}\Big]\mathbb{E}\Big[\boldsymbol{f}^\top\Big] \cancel{-\mathbb{E}\Big[\boldsymbol{f}\Big]\mathbb{E}\Big[\boldsymbol{f}^\top\Big]} \cancel{+\mathbb{E}\Big[\boldsymbol{f}\Big]\mathbb{E}\Big[\boldsymbol{f}^\top\Big]} \\
\mathbb{E}\Big[\boldsymbol{f}\boldsymbol{f}^\top\Big] &= \mathrm{cov}[\boldsymbol{f}, \boldsymbol{f}^\top] + \mathbb{E}\Big[\boldsymbol{f}\Big]\mathbb{E}\Big[\boldsymbol{f}^\top\Big] \\
\mathbb{E}_{p(\boldsymbol{f}|\mu,\Sigma)}\Big[\boldsymbol{f}\boldsymbol{f}^\top\Big] &= \mathrm{cov}[\boldsymbol{f}, \boldsymbol{f}^\top] + \mathbb{E}_{p(\boldsymbol{f}|\mu,\Sigma)}\Big[\boldsymbol{f}\Big]\mathbb{E}_{p(\boldsymbol{f}|\mu,\Sigma)}\Big[\boldsymbol{f}^\top\Big] \\
\mathbb{E}_{p(\boldsymbol{f}|\mu,\Sigma)}\Big[\boldsymbol{f}\boldsymbol{f}^\top\Big] &= \Sigma + \mu\mu^\top
\end{aligned} \tag{A.14}$$

Appendix B

Kernels

In Section 3.1.1 a number of common kernel functions were discussed, and samples were provided for a number of others. This section outlines the form of each of these additional kernels.

B.0.1 Polynomial

The polynomial uses polynomial basis functions of a certain order, o.

$$K(\mathbf{x}, \mathbf{x}') = \sigma^2(\alpha \mathbf{x}^\top \mathbf{x}' + b)^o \tag{B.1}$$

where α is the scale, σ^2 is the variance, and b is the bias.

B.0.2 Matern 32 and Matern 52

The Matern family of kernels are very common in practical applications, as it is possible to vary the number of times they are differentiable by different choices in a parameter. The most common choices are $\frac{3}{2}$,

$$K(\mathbf{x}, \mathbf{x}') = \sigma^2(1 + \sqrt{3}r)\exp(-\sqrt{3}r) \tag{B.2}$$

and $\frac{5}{2}$,

$$K(\mathbf{x}, \mathbf{x}') = \sigma^2(1 + \sqrt{5}r + \frac{5}{3}r^2)\exp(-\sqrt{5}r) \tag{B.3}$$

where σ^2 is the variance, ℓ is the lengthscale, and $r = \sqrt{\sum_{i=1}^{q} \frac{(x_i - x_i')^2}{\ell_i^2}}$

B.0.3 Brownian

The Brownian kernel simulates the path of a random walk and is not differentiable. It is given by,

$$K(\mathbf{x}, \mathbf{x}') = \sigma^2 \mathrm{min}(\mathbf{x}, \mathbf{x}') \tag{B.4}$$

where σ^2 is the variance.

B.0.4 Bias

The bias kernel is a very simple kernel that puts a prior over where the latent mean function is (usually assumed to be zero),

$$K(\mathbf{x}, \mathbf{x}') = \sigma^2 \tag{B.5}$$

where the variance is σ^2

Appendix C

Chained Gaussian Process Supplementary Material

In this chapter some additonal details for some aspects of the chained Gaussian process model introduced in Chapter 4 are provided, including the method by which gradients can be computed and additional results. These are primarily reproductions of the supplementary material for the original paper, Saul et al. (2016). These will be referred to throughout Chapter 4.

C.1 Gradients and Optimisation

Gradients can be computed similarly to (Hensman et al., 2015b) using the equalities,

$$\frac{\partial}{\partial \mu}\mathbb{E}_{\mathcal{N}(x|\mu,\sigma^2)}\left[f(x)\right] = \mathbb{E}_{\mathcal{N}(x|\mu,\sigma^2)}\left[\frac{\partial}{\partial x}f(x)\right] \tag{C.1}$$

$$\frac{\partial}{\partial \sigma^2}\mathbb{E}_{\mathcal{N}(x|\mu,\sigma^2)}\left[f(x)\right] = \frac{1}{2}\mathbb{E}_{\mathcal{N}(x|\mu,\sigma^2)}\left[\frac{\partial}{\partial x^2}f(x)\right] \tag{C.2}$$

and the chain rule.

Since our posterior assumes factorisation between $q(\boldsymbol{f})$ and $q(\boldsymbol{g})$ the gradients can be computed independently using (C.1) and (C.2),

$$\begin{gathered}\frac{\partial}{\partial \boldsymbol{\mu}_f}\mathbb{E}_{\mathcal{N}\left(\mathbf{x}_i|\boldsymbol{m}_f,\boldsymbol{v}_f\right)}\left[\log p(\mathbf{y}|\boldsymbol{f},\boldsymbol{g})\right]\\ \frac{\partial}{\partial \boldsymbol{\mu}_g}\mathbb{E}_{\mathcal{N}(\mathbf{x}_i|\boldsymbol{m}_g,\boldsymbol{v}_g)}\left[\log p(\mathbf{y}|\boldsymbol{f},\boldsymbol{g})\right]\\ \frac{\partial}{\partial \boldsymbol{v}_f}\mathbb{E}_{\mathcal{N}\left(\mathbf{x}_i|\boldsymbol{m}_f,\boldsymbol{v}_f\right)}\left[\log p(\mathbf{y}|\boldsymbol{f},\boldsymbol{g})\right]\\ \frac{\partial}{\partial \boldsymbol{v}_g}\mathbb{E}_{\mathcal{N}(\mathbf{x}_i|\boldsymbol{m}_g,\boldsymbol{v}_g)}\left[\log p(\mathbf{y}|\boldsymbol{f},\boldsymbol{g})\right],\end{gathered}$$

The expectations can then be done using quadrature, or Monte Carlo sampling. As before,

$$\begin{aligned}\boldsymbol{m}_f &= \mathbf{K}_{\boldsymbol{f}\mathbf{u}}\mathbf{K}^{-1}_{\mathbf{u}_f\mathbf{u}_f}\boldsymbol{\mu}_f\\ \boldsymbol{v}_f &= \mathbf{K}_{\boldsymbol{f}\boldsymbol{f}} + \mathbf{K}_{\boldsymbol{f}\mathbf{u}}\mathbf{K}^{-1}_{\mathbf{u}_f\mathbf{u}_f}(\boldsymbol{S}_f - \mathbf{K}_{\mathbf{u}_f\mathbf{u}_f})\mathbf{K}^{-1}_{\mathbf{u}_f\mathbf{u}_f}\mathbf{K}_{\mathbf{u}\boldsymbol{f}}\\ \boldsymbol{m}_g &= \mathbf{K}_{\boldsymbol{g}\mathbf{u}_g}\mathbf{K}^{-1}_{\mathbf{u}_g\mathbf{u}_g}\boldsymbol{\mu}_g\\ \boldsymbol{v}_g &= \mathbf{K}_{\boldsymbol{g}\boldsymbol{g}} + \mathbf{K}_{\boldsymbol{g}\mathbf{u}_g}\mathbf{K}^{-1}_{\mathbf{u}_g\mathbf{u}_g}(\boldsymbol{S}_g - \mathbf{K}_{\mathbf{u}_g\mathbf{u}_g})\mathbf{K}^{-1}_{\mathbf{u}_g\mathbf{u}_g}\mathbf{K}_{\mathbf{u}_g\boldsymbol{g}}.\end{aligned}$$

The chain rule can then be used to compute the gradient with respect to the kernel hyper parameters, $\frac{\partial}{\partial \boldsymbol{m}_f}\mathbb{E}_{\mathcal{N}\left(\mathbf{x}_i|\boldsymbol{m}_f,\boldsymbol{v}_f\right)}\left[\log p(\mathbf{y}|\boldsymbol{f},\boldsymbol{g})\right]\frac{\partial \boldsymbol{m}_f}{\partial \mathbf{K}_{\boldsymbol{f}\mathbf{u}}}\frac{\partial \mathbf{K}_{\boldsymbol{f}\mathbf{u}}}{\partial \boldsymbol{\theta}_K}$, where $\boldsymbol{\theta}_K$ is a hyper parameter of the kernel $k_{\boldsymbol{f}}$. Similar chain rules can be written for the other derivatives.

The model contains variational parameters corresponding to $q(\mathbf{u}_f) = \mathcal{N}(\mathbf{u}_f|\boldsymbol{\mu}_f, \boldsymbol{S}_f)$ and $q(\mathbf{u}_g) = \mathcal{N}(\mathbf{u}_g|\boldsymbol{\mu}_g, \boldsymbol{S}_g)$ and the latent input locations, $\mathbf{Z}$. As such the parameters do not scale with n. Naively the number of parameters is $\mathcal{O}(b(m^2+m)+m)$ however we can reduce this to $\mathcal{O}(b(\frac{m^2}{2}+\frac{m}{2}+m))$ by parametrizing the Cholesky of the covariance matrices, $\boldsymbol{S}_f = L_{\boldsymbol{f}}L_{\boldsymbol{f}}^{\top}$ and $\boldsymbol{S}_g = L_{\boldsymbol{g}}L_{\boldsymbol{g}}^{\top}$. This has the added benefit of enforcing that $\boldsymbol{S}_f$ and $\boldsymbol{S}_g$ are symmetrical and positive definite. Unfortunately if there is a large number of inducing inputs, this can be difficult to optimise.

In order to aid future users, we provide some advice on our tactic of optimising these models. We initialize the model with random or informed lengthscales within the right region, $\boldsymbol{\mu}_f$ and $\boldsymbol{\mu}_g$ are assigned small random values, $\boldsymbol{S}_g$ and $\boldsymbol{S}_f$ are given an identity form. In practice during optimisation we find it helpful to initially fix all the kernel hyperparameters and $\mathbf{Z}$ at their initial locations, optimise for a small number of steps, then allow the optimisation to run freely. This allows the latent means $\boldsymbol{\mu}_f$ and $\boldsymbol{\mu}_g$ to move to sensible locations before the model is allowed to completely change the

Data	**MAE**			
	G	**CHG**	**Lt**	**CHt**
Elevators1000	0.252	0.272	-	-
Elevators10000	0.219	0.224	-	-
CorruptMotor	0.947	0.841	0.835	0.834
Boston	0.048	0.050	0.050	0.050

Table C.1 Results showing the MAE and NLPD over 5 cross validation folds, Models shown in comparison are sparse Gaussian (G), chained heteroscedastic Gaussian (CHG), Student-t Laplace approximation (Lt) and chained heteroscedastic Student-t (CHt).

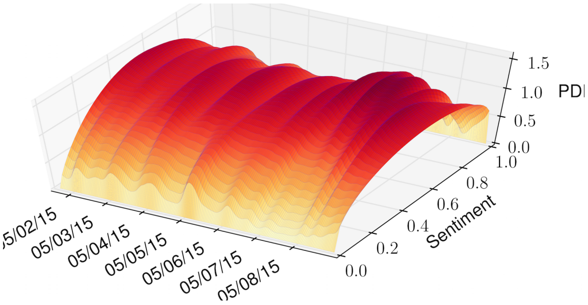
form of the function through the modification of the kernel hyperparameters. True convergence can be difficult to achieve due to the potentially large number of strongly dependent parameters and the non-convex optimisation problem, and in practice we find it helpful to monitor convergence. There is a particularly strong dependence between the inducing inputs $\mathbf{Z}$ and the variational parameters; we recommend fixing $\mathbf{Z}$ when they are no longer drastically moving. Despite challenges, it is important to note however that the number of parameters to be optimised is *fixed* with respect to n, and stochastic methods can be used to allow consideration of a very large n.

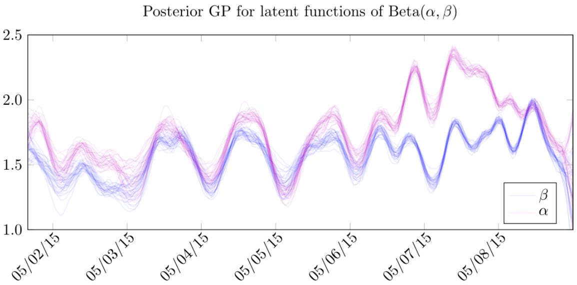
C.2 Additional MAE Results

Alongside negative log predictive density (NLPD) we also measure MAE. In practice we find less significant changes in the MAE as the NLPD, particularly for the testing of robust methods. This indicates that the most significant increase in performance we gain is the ability to explain the tails, where the outliers live; Gaussian processes are particularly useful for taking into consideration the uncertainty associated with predictions. Small differences between MAE are likely to be attributed to the mean function being pulled towards these outliers. However, when testing robustness we hope that the mean is *not* pulled towards the outlying data. As such we reason that NLPD is a better measure to test robustness.

C.3 Further Twitter Experiment Details

The model used to model the twitter data has some interesting properties, such as the ability to model a transition from a unimodal distribution to a bimodal distribution. The following plot shows how the distribution changes throughout time for the Labour dataset.




Posterior GP for latent functions of Beta(α, β)

The latent functions α and β that can be obtained from the model in Section 4.4.3 can be plotted themselves. If both latent functions went below 1.0 then the distribution at that time would turn into a bathtub shape. If both are larger than one but one is larger than the other, we have a skewed distribution. If one is below zero and the over above, it appears exponential or negative exponential.

Appendix D

Bayesian Gaussian Process Latent Variable Model Details

In order to maintain clarity in the main text we choose not to derive the BGPLVM for missing data model and LABGPLVM model from scratch. However various equations from the derivation are required for the main text; and need to be referenced accordingly. Here a full derivation from a generative modelling perspective is produced for a particularly interested reader. It is hoped that the step by step derivation of a relatively complex model is useful for those interested in building upon the research in Chapter 5. Also see Damianou (2015) and Gal and van der Wilk (2014) for excellent similarly detailed derivations.

D.1 BGPLVM Generative Model and Overview of Derivation

First an overview of the model and the lower bounds required are given.

The generative model for the BGPLVM is as follows,

$$p(\mathbf{Y}, \boldsymbol{F}, \mathbf{U}, \mathbf{X}, |\mathbf{Z}) = p(\mathbf{Y}|\boldsymbol{F})p(\boldsymbol{F}|\mathbf{U}, \mathbf{X})p(\mathbf{U}|\mathbf{Z})p(\mathbf{X}), \tag{D.1}$$

where

$$\begin{aligned}
p(\mathbf{Y}|\boldsymbol{F}) &= \prod_{j=1}^{p} \mathcal{N}\left(\mathbf{y}_{:,j}|\boldsymbol{f}_{:,j}, \beta^{-1}\mathbf{I}\right) \\
p(\boldsymbol{F}|\mathbf{U}, \mathbf{X}) &= \prod_{j=1}^{p} \mathcal{N}\left(\boldsymbol{f}_{:,j}|\mathbf{K}_{\boldsymbol{f}\mathbf{u}}\mathbf{K}_{\mathbf{uu}}^{-1}\mathbf{u}_{:,j}, \mathbf{K}_{\boldsymbol{f}\boldsymbol{f}} - \mathbf{K}_{\boldsymbol{f}\mathbf{u}}\mathbf{K}_{\mathbf{uu}}^{-1}\mathbf{K}_{\mathbf{u}\boldsymbol{f}}\right) \\
p(\mathbf{u}_{:,j}|\mathbf{Z}) &= \mathcal{N}\left(\mathbf{u}_{:,j}|\mathbf{0}, \mathbf{K}_{\mathbf{uu}}\right) \\
p(\mathbf{X}) &= \mathcal{N}\left(\mathbf{X}|\mathbf{0}, \mathbf{I}\right).
\end{aligned}$$

Since the $p(\mathbf{Y}|\boldsymbol{F}$ and $p(\boldsymbol{f}_{:,j}|\mathbf{u}_{:,j}, \mathbf{X})$ factorise across output dimensions, and these two functions are the only two functions that live in the data space, the marginal likelihood also factorises,

$$\begin{aligned}
\log p(\mathbf{y}|\mathbf{Z}) &= \log \prod_{j=1}^{p} p(\mathbf{y}_j|\mathbf{Z}) \\
&= \sum_{j=1}^{p} \log p(\mathbf{y}_j|\mathbf{Z}),
\end{aligned}$$

In what follows the variational bound derived will be expressed in terms of a single dimension $p(\mathbf{y}_{:,j}|\mathbf{Z})$ that will subsequently be summed together, each will be denoted with a *subscript* j, for example is $p(\mathbf{y}_{:,j}|\mathbf{Z})$. A number of *variational lower bounds* will be derived, when a distribution refers to a lower bound on the true distribution, it will be denoted by a $\hat{p}$; for example $\hat{p}_j(\mathbf{y}_{:,j}|\mathbf{Z})$ is a variational lower bound on $p(\mathbf{y}_{:,j}|\mathbf{Z})$. For the missing data case, you can consider $\mathbf{y}_{:,j}$ as only the non-missing values of $\mathbf{y}_{:,j}$, i.e. $\mathbf{y}_{:,j}^{\mathcal{O}_j}$ in the main text. Analogically $\boldsymbol{\psi}_1$, $\boldsymbol{\psi}_2$, ψ_0, $\mathbf{y}_{:,j}$, $\hat{\mathbf{y}}_{:,j}$, $\hat{\mathbf{K}}$, and $\boldsymbol{W}_j$ would be replaced by ${\boldsymbol{\psi}_1}^{\mathcal{O}_j}$, ${\boldsymbol{\psi}_2}^{\mathcal{O}_j}$, ${\psi_0}^{\mathcal{O}_j}$, $\mathbf{y}_{:,j}^{\mathcal{O}_j}$, $\hat{\mathbf{y}}_{:,j}^{\mathcal{O}_j}$, $\hat{\mathbf{K}}^{\mathcal{O}_j}$ and $\boldsymbol{W}^{\mathcal{O}_j}$.

The following is a small sketch of how we derive the final bound followed by a more in-depth treatment.

$$\log p(\mathbf{y}_{:,j}|\mathbf{Z}) = \log \int\int\int p(\mathbf{y}_{:,j}|\boldsymbol{f}_{:,j})p(\boldsymbol{f}_{:,j}|\mathbf{u}_{:,j},\mathbf{X})p(\mathbf{u}_{:,j}|\mathbf{Z})p(\mathbf{X})\, d\boldsymbol{f}_{:,j}\, d\mathbf{u}_{:,j}\, d\mathbf{X} \tag{D.2}$$

$$= \log \int\int \exp\left[\log\left(\int p(\mathbf{y}_{:,j}|\boldsymbol{f}_{:,j})p(\boldsymbol{f}_{:,j}|\mathbf{u}_{:,j},\mathbf{X})\, d\boldsymbol{f}_{:,j}\right)\right] p(\mathbf{u}_{:,j}|\mathbf{Z})p(\mathbf{X})\, d\mathbf{u}_{:,j}\, d\mathbf{X} \tag{D.3}$$

$$\geq \log \int\int \exp\left[\underbrace{\int p(\boldsymbol{f}_{:,j}|\mathbf{u}_{:,j},\mathbf{X}) \log p(\mathbf{y}_{:,j}|\boldsymbol{f}_{:,j})\, d\boldsymbol{f}_{:,j}}_{\hat{p}_j(\mathbf{y}_{:,j}|\mathbf{u}_{:,j},\mathbf{X})}\right] p(\mathbf{u}_{:,j}|\mathbf{Z})p(\mathbf{X})\, d\mathbf{u}_{:,j}\, d\mathbf{X} \tag{D.4}$$

$$= \log \int\int \exp(\log(\hat{p}_j(\mathbf{y}_{:,j}|\mathbf{u}_{:,j},\mathbf{X})))p(\mathbf{u}_{:,j}|\mathbf{Z})p(\mathbf{X})\, d\mathbf{X}\, d\mathbf{u}_{:,j} \tag{D.5}$$

$$= \log \int\int \exp(\log(\hat{p}_j(\mathbf{y}_{:,j}|\mathbf{u}_{:,j},\mathbf{X})))\frac{q(\mathbf{X}|\boldsymbol{\theta}_V)}{q(\mathbf{X}|\boldsymbol{\theta}_V)}p(\mathbf{X})p(\mathbf{u}_{:,j}|\mathbf{Z})\, d\mathbf{u}_{:,j}\, d\mathbf{X} \tag{D.6}$$

$$\geq \int \left[\int \log \exp(\log(\hat{p}_j(\mathbf{y}_{:,j}|\mathbf{u}_{:,j},\mathbf{X})))q(\mathbf{X}|\boldsymbol{\theta}_V)\, d\mathbf{X}\right] p(\mathbf{u}_{:,j}|\mathbf{Z})\, d\mathbf{u}_{:,j} \tag{D.7}$$

$$- \mathrm{KL}\left(q(\mathbf{X}|\boldsymbol{\theta}_V) \,\|\, p(\mathbf{X})\right) \tag{D.8}$$

$$= \int \left[\int q(\mathbf{X}|\boldsymbol{\theta}_V) \log(\hat{p}_j(\mathbf{y}_{:,j}|\mathbf{u}_{:,j},\mathbf{X}))\, d\mathbf{X}\right] p(\mathbf{u}_{:,j}|\mathbf{Z})\, d\mathbf{u}_{:,j} - \mathrm{KL}\left(q(\mathbf{X}|\boldsymbol{\theta}_V) \,\|\, p(\mathbf{X})\right) \tag{D.9}$$

$$= \left[\int q(\mathbf{X}|\boldsymbol{\theta}_V) \log(\hat{p}_j(\mathbf{y}_{:,j}|\mathbf{X}))\, d\mathbf{X}\right] - \mathrm{KL}\left(q(\mathbf{X}|\boldsymbol{\theta}_V) \,\|\, p(\mathbf{X})\right) \tag{D.10}$$

D.2 Integrating Out $p(\boldsymbol{f}_{:,j}|\mathbf{u}_{:,j},\mathbf{X})$

In Equation:(D.4) Jensen's inequality is used, such that

$$\log \mathbb{E}_{p(\boldsymbol{f}_{:,j}|\mathbf{u}_{:,j},\mathbf{X})}\Big[p(\mathbf{y}_{:,j}|\mathbf{u}_{:,j},\boldsymbol{f}_{:,j},\mathbf{X})\Big] \geq \mathbb{E}_{p(\boldsymbol{f}_{:,j}|\mathbf{u}_{:,j},\mathbf{X})}\Big[\log p(\mathbf{y}_{:,j}|\mathbf{u}_{:,j},\boldsymbol{f}_{:,j},\mathbf{X})\Big].$$

The left hand side is intractable, however the right hand side is tractable when $p(\mathbf{y}_{:,j}|\mathbf{u}_{:,j},\boldsymbol{f}_{:,j},\mathbf{X})$ is a Gaussian likelihood.

In more detail,

$$\begin{aligned}\log p(\mathbf{y}_{:,j}|\mathbf{X},\mathbf{u}_{:,j}) &= \log \int p(\mathbf{y}_{:,j}|\boldsymbol{f}_{:,j})p(\boldsymbol{f}_{:,j}|\mathbf{u}_{:,j},\mathbf{X})\, d\boldsymbol{f}_{:,j}\\
&= \log \mathbb{E}_{p(\boldsymbol{f}_{:,j}|\mathbf{u}_{:,j},\mathbf{X})}\Big[p(\mathbf{y}_{:,j}|\boldsymbol{f}_{:,j})\Big]\\
&\geq \mathbb{E}_{p(\boldsymbol{f}_{:,j}|\mathbf{u}_{:,j},\mathbf{X})}\Big[\log p(\mathbf{y}_{:,j}|\boldsymbol{f}_{:,j})\Big]\\
&= \mathbb{E}_{p(\boldsymbol{f}_{:,j}|\mathbf{u}_{:,j},\mathbf{X})}\Big[-\frac{n}{2}\log 2\pi - \frac{1}{2}\log|\beta^{-1}\mathbf{I}| + \frac{\beta}{2}(\mathbf{y}_{:,j}-\boldsymbol{f}_{:,j})^\top(\mathbf{y}_{:,j}-\boldsymbol{f}_{:,j})\Big]\\
&= \mathbb{E}_{p(\boldsymbol{f}_{:,j}|\mathbf{u}_{:,j},\mathbf{X})}\Big[-\frac{n}{2}\log 2\pi - \frac{1}{2}\log|\beta^{-1}\mathbf{I}| + \frac{\beta}{2}\mathrm{tr}\left((\mathbf{y}_{:,j}-\boldsymbol{f}_{:,j})^\top(\mathbf{y}_{:,j}-\boldsymbol{f}_{:,j})\right)\Big]\\
&= -\frac{n}{2}\log 2\pi - \frac{1}{2}\log|\beta^{-1}\mathbf{I}|\\
&\quad + \frac{\beta}{2}\mathrm{tr}\left(\mathbf{y}_{:,j}^\top\mathbf{y}_{:,j} - 2\mathbf{y}_{:,j}^\top\mathbb{E}_{p(\boldsymbol{f}_{:,j}|\mathbf{u}_{:,j},\mathbf{X})}\Big[\boldsymbol{f}_{:,j}\Big] + \mathbb{E}_{p(\boldsymbol{f}_{:,j}|\mathbf{u}_{:,j},\mathbf{X})}\Big[\boldsymbol{f}_{:,j}\boldsymbol{f}_{:,j}^\top\Big]\right),\end{aligned} \tag{D.11}$$

where in the final line the trace is used for $\mathrm{tr}\left(\boldsymbol{f}_{:,j}\boldsymbol{f}_{:,j}^\top\right) = \mathrm{tr}\left(\boldsymbol{f}_{:,j}^\top\boldsymbol{f}_{:,j}\right)$.

Since the first moment is the expected value of the distribution, $\mathbb{E}_{p(\boldsymbol{f}_{:,j}|\mathbf{u}_{:,j},\mathbf{X})}\Big[\boldsymbol{f}_{:,j}\Big]$ is the mean of the distribution of $p(\boldsymbol{f}_{:,j}|\mathbf{u}_{:,j},\mathbf{X})$, in this case $\mathbf{K}_{\boldsymbol{f}\mathbf{u}}\mathbf{K}_{\mathbf{uu}}^{-1}\mathbf{u}_{:,j}$. Using the equality in Equation (A.14), $\mathrm{cov}[\boldsymbol{f}_{:,j},\boldsymbol{f}_{:,j}^\top]$ is the covariance of $p(\boldsymbol{f}_{:,j}|\mathbf{u}_{:,j},\mathbf{X})$, in this case $\mathbf{K}_{\boldsymbol{ff}} - \mathbf{K}_{\boldsymbol{f}\mathbf{u}}\mathbf{K}_{\mathbf{uu}}^{-1}\mathbf{K}_{\mathbf{u}\boldsymbol{f}}$ giving

$$\mathbb{E}\Big[\boldsymbol{f}_{:,j}\boldsymbol{f}_{:,j}^\top\Big] = \mathbf{K}_{\boldsymbol{ff}} - \mathbf{K}_{\boldsymbol{f}\mathbf{u}}\mathbf{K}_{\mathbf{uu}}^{-1}\mathbf{K}_{\mathbf{u}\boldsymbol{f}} + \mathbf{K}_{\boldsymbol{f}\mathbf{u}}\mathbf{K}_{\mathbf{uu}}^{-1}\mathbf{u}_{:,j}\mathbf{u}_{:,j}^\top\mathbf{K}_{\mathbf{uu}}^{-1}\mathbf{K}_{\mathbf{u}\boldsymbol{f}} \tag{D.12}$$

By substituting the above into Equation (D.11), and completing the square with respect to $\mathbf{y}_{:,j}$, we obtain a log of a Gaussian distribution with some additional terms,

$$\begin{aligned}\log p(\mathbf{y}_{:,j}|\mathbf{X},\mathbf{u}_{:,j}) &= +\mathrm{KL}\left(p(\boldsymbol{f}|\mathbf{u},\mathbf{X})\parallel p(\boldsymbol{f}|\mathbf{y},\mathbf{u},\mathbf{X})\right) - \frac{n}{2}\log 2\pi - \frac{1}{2}\log|\beta^{-1}\mathbf{I}|\\
&\quad - \frac{\beta}{2}\mathrm{tr}\left(\mathbf{y}_{:,j}^\top\mathbf{y}_{:,j} - 2\mathbf{y}_{:,j}^\top\mathbf{K}_{\boldsymbol{f}\mathbf{u}}\mathbf{K}_{\mathbf{uu}}^{-1}\mathbf{u}_{:,j}\right.\\
&\qquad \left.+\mathbf{K}_{\boldsymbol{ff}} - \mathbf{K}_{\boldsymbol{f}\mathbf{u}}\mathbf{K}_{\mathbf{uu}}^{-1}\mathbf{K}_{\mathbf{u}\boldsymbol{f}} + \mathbf{K}_{\boldsymbol{f}\mathbf{u}}\mathbf{K}_{\mathbf{uu}}^{-1}\mathbf{u}_{:,j}\mathbf{u}_{:,j}^\top\mathbf{K}_{\mathbf{uu}}^{-1}\mathbf{K}_{\mathbf{u}\boldsymbol{f}}\right)\\
&\geq -\frac{n}{2}\log 2\pi - \frac{1}{2}\log|\beta^{-1}\mathbf{I}|\\
&\quad - \frac{\beta}{2}\mathrm{tr}\left(\mathbf{y}_{:,j}^\top\mathbf{y}_{:,j} - 2\mathbf{y}_{:,j}^\top\mathbf{K}_{\boldsymbol{f}\mathbf{u}}\mathbf{K}_{\mathbf{uu}}^{-1}\mathbf{u}_{:,j}\right.\\
&\qquad \left.+\mathbf{K}_{\boldsymbol{ff}} - \mathbf{K}_{\boldsymbol{f}\mathbf{u}}\mathbf{K}_{\mathbf{uu}}^{-1}\mathbf{K}_{\mathbf{u}\boldsymbol{f}} + \mathbf{K}_{\boldsymbol{f}\mathbf{u}}\mathbf{K}_{\mathbf{uu}}^{-1}\mathbf{u}_{:,j}\mathbf{u}_{:,j}^\top\mathbf{K}_{\mathbf{uu}}^{-1}\mathbf{K}_{\mathbf{u}\boldsymbol{f}}\right)\end{aligned}$$

Complete square, $\mu = \mathbf{K}_{\boldsymbol{f}\mathbf{u}}\mathbf{K}_{\mathbf{uu}}^{-1}\mathbf{u}_{:,j}, \Sigma = \beta^{-1}\mathbf{I}$

$$
\begin{aligned}
&= \log \mathcal{N}\left(\mathbf{y}_{:,j}|\mathbf{K}_{\boldsymbol{f}\mathbf{u}}\mathbf{K}_{\mathbf{uu}}^{-1}\mathbf{u}_{:,j}, \beta^{-1}\mathbf{I}\right) - \frac{\beta}{2}\mathrm{tr}\left(\mathbf{K}_{\boldsymbol{ff}} - \mathbf{K}_{\boldsymbol{f}\mathbf{u}}\mathbf{K}_{\mathbf{uu}}^{-1}\mathbf{K}_{\mathbf{u}\boldsymbol{f}}\right) \\
&= \log \hat{p}_j(\mathbf{y}_{:,j}|\mathbf{X}, \mathbf{u}_{:,j})
\end{aligned}
$$

The term $\frac{\beta}{2}\mathrm{tr}\left(\mathbf{K}_{\boldsymbol{ff}} - \mathbf{K}_{\boldsymbol{f}\mathbf{u}}\mathbf{K}_{\mathbf{uu}}^{-1}\mathbf{K}_{\mathbf{u}\boldsymbol{f}}\right)$ is from now on referred to as the *trace term*. This term guards the bound from overfitting (Damianou, 2015; Titsias, 2009). It corresponds to the squared error of predicting the training values of $\boldsymbol{f}_{:,j}$ using the inducing variables $\mathbf{u}_{:,j}$. Remembering that the number of inducing variables m, is chosen to be smaller than the number of latent function variables n, as in Section 3.3; if the number of inducing variables are insufficient for encoding the information that would otherwise be held in $\boldsymbol{f}_{:,j}$, then this term will be large, and the lower bound will become loose. Note, that by assuming a low-rank Nystrom approximation, $p(\boldsymbol{f}_{:,j}|\mathbf{u}_{:,j}, \mathbf{X}) = \mathcal{N}\left(\boldsymbol{f}_{:,j}|\mathbf{K}_{\boldsymbol{f}\mathbf{u}}\mathbf{K}_{\mathbf{uu}}^{-1}\mathbf{u}_{:,j}, \mathbf{K}_{\boldsymbol{ff}} - \mathbf{K}_{\boldsymbol{f}\mathbf{u}}\mathbf{K}_{\mathbf{uu}}^{-1}\mathbf{K}_{\mathbf{u}\boldsymbol{f}}\right)$, as is also used in the sparse GP regression case, Section 3.3, the form of $\log \hat{p}_j(\mathbf{y}|\mathbf{u}, \mathbf{X})$, does not contain the inverse of a kernel function, $k(\mathbf{X}, \mathbf{X})^{-1}$ that operates on the inputs $\mathbf{X}$. This correspondingly reduces the computational complexity from $\mathcal{O}(n^3)$ to $\mathcal{O}(nm^2)$. In the following section it will also become apparent that this allows use to use variational inference to obtain an approximation to the bound, as expectations can now be taken with respect to $\mathbf{X}$ without requiring the intractable expectation of an inverse.

D.2.1 Integration Over $p(\mathbf{X})$

The next step to reaching the final bound on $\log p(\mathbf{y}_{:,j}|\mathbf{Z})$ is to integrate out $p(\mathbf{X})$. As in Equation (D.5) the bound $\log \hat{p}_j(\mathbf{y}_{:,j}|\mathbf{X}, \mathbf{u}_{:,j})$, will be substituted for $\log p(\mathbf{y}_{:,j}|\mathbf{X}, \mathbf{u}_{:,j})$, thus the next step is,

$$
\begin{aligned}
\log p(\mathbf{y}_{:,j}|\mathbf{Z}, \mathbf{u}) &= \log \int p(\mathbf{y}_{:,j}|\mathbf{X}, \mathbf{u}_{:,j})p(\mathbf{X})\, d\mathbf{X} \\
&\geq \log \int \exp(\log \hat{p}_j(\mathbf{y}_{:,j}|\mathbf{X}, \mathbf{u}_{:,j}))p(\mathbf{X})\, d\mathbf{X} \\
&= \log \hat{p}_j(\mathbf{y}_{:,j}|\mathbf{Z}, \mathbf{u}).
\end{aligned}
$$

Here the expectation is with respect to a kernel $\mathbf{K}_{\boldsymbol{f}\mathbf{u}}$ that is in an exponent. As alluded to above, this integration can be done using the variational trick, of introducing $q(\mathbf{X}|\boldsymbol{\theta}_V)$. By introducing the sparse variational inference approach, Section 3.3, we have ensured that the expectation does not need to be taken over the inverse of a kernel $\mathbf{K}(\mathbf{X},\mathbf{X})^{-1}$, which is intractable, as discussed in Section 5.5.

The next step is to compute $\log \hat{p}_j(\mathbf{y}_{:,j}|\mathbf{u}_{:,j}) = \log \mathbb{E}_{q(\mathbf{X}|\boldsymbol{\theta}_V)}\left[\frac{\hat{p}_j(\mathbf{y}_{:,j}|\mathbf{u}_{:,j},\mathbf{X})p(\mathbf{X})}{q(\mathbf{X}|\boldsymbol{\theta}_V)}\right]$, which may be done by introducing a variational distribution $q(\mathbf{X}|\boldsymbol{\theta}_V)$, as follows,

$$
\begin{aligned}
\log p(\mathbf{y}_{:,j}|\mathbf{u}_{:,j}) &= \log \mathbb{E}_{q(\mathbf{X}|\boldsymbol{\theta}_V)}\left[\frac{p(\mathbf{y}_{:,j}|\mathbf{u}_{:,j},\mathbf{X})p(\mathbf{X})}{q(\mathbf{X}|\boldsymbol{\theta}_V)}\right] \\
&= \mathbb{E}_{q(\mathbf{X}|\boldsymbol{\theta}_V)}\left[\log \hat{p}_j(\mathbf{y}_{:,j}|\mathbf{u}_{:,j},\mathbf{X})\frac{p(\mathbf{X})}{q(\mathbf{X}|\boldsymbol{\theta}_V)}\right] + \mathrm{KL}\left(q(\mathbf{X}|\boldsymbol{\theta}_V)\,\|\,p(\mathbf{X}|\mathbf{y})\right) \\
&\geq \mathbb{E}_{q(\mathbf{X}|\boldsymbol{\theta}_V)}\left[\log \hat{p}_j(\mathbf{y}_{:,j}|\mathbf{u}_{:,j},\mathbf{X})\frac{p(\mathbf{X})}{q(\mathbf{X}|\boldsymbol{\theta}_V)}\right] \\
&= \mathbb{E}_{q(\mathbf{X}|\boldsymbol{\theta}_V)}\left[\log \hat{p}_j(\mathbf{y}_{:,j}|\mathbf{u}_{:,j},\mathbf{X})\right] + \mathbb{E}_{q(\mathbf{X}|\boldsymbol{\theta}_V)}\left[\log \frac{p(\mathbf{X})}{q(\mathbf{X}|\boldsymbol{\theta}_V)}\right] \\
&= \underbrace{\mathbb{E}_{q(\mathbf{X}|\boldsymbol{\theta}_V)}\left[\log \hat{p}_j(\mathbf{y}_{:,j}|\mathbf{u}_{:,j},\mathbf{X})\right]}_{\triangleq \log \hat{p}_j(\mathbf{y}_{:,j}|\mathbf{u}_{:,j})} - \mathrm{KL}\left(q(\mathbf{X}|\boldsymbol{\theta}_V)\,\|\,p(\mathbf{X})\right) \\
&= \log \hat{p}_j(\mathbf{y}_{:,j}|\mathbf{u}_{:,j}) - \mathrm{KL}\left(q(\mathbf{X}|\boldsymbol{\theta}_V)\,\|\,p(\mathbf{X})\right).
\end{aligned}
$$

Note that now an expectation of the log of the previous bound can be taken, with respect to the variational distribution $q(\mathbf{X}|\boldsymbol{\theta}_V)$, $\mathbb{E}_{q(\mathbf{X}|\boldsymbol{\theta}_V)}\left[\log \hat{p}_j(\mathbf{y}_{:,j}|\mathbf{u}_{:,j},\mathbf{X})\right]$, since $\hat{p}_j(\mathbf{y}_{:,j}|\mathbf{u}_{:,j},\mathbf{X})$ is Gaussian, this is now analytically tractable,

$$
\begin{aligned}
&\mathbb{E}_{q(\mathbf{X}|\boldsymbol{\theta}_V)}\left[\log \hat{p}_j(\mathbf{y}_{:,j}|\mathbf{u}_{:,j},\mathbf{X})\right] \\
&= \mathbb{E}_{q(\mathbf{X}|\boldsymbol{\theta}_V)}\left[-\frac{n}{2}\log 2\pi - \frac{1}{2}\log|\beta^{-1}\mathbf{I}_j| - \frac{1}{2}\left(\mathbf{y}_{:,j}^{\top}\beta\mathbf{I}_j\mathbf{y}_{:,j} - 2\mathbf{y}_{:,j}^{\top}\beta\mathbf{I}_j\mathbf{K}_{\boldsymbol{f}\mathbf{u}}\mathbf{K}_{\mathbf{uu}}^{-1}\mathbf{u}_{:,j}\right.\right. \qquad \text{(D.13)} \\
&\left.\left.+\mathbf{u}_{:,j}^{\top}\mathbf{K}_{\mathbf{uu}}^{-1}\mathbf{K}_{\mathbf{u}\boldsymbol{f}}\beta\mathbf{I}_j\mathbf{K}_{\boldsymbol{f}\mathbf{u}}\mathbf{K}_{\mathbf{uu}}^{-1}\mathbf{u}_{:,j}\right)\right] - \mathbb{E}_{q(\mathbf{X}|\boldsymbol{\theta}_V)}\left[\frac{\beta}{2}\mathrm{tr}\left(\mathbf{K}_{\boldsymbol{ff}} - \mathbf{K}_{\mathbf{uu}}^{-1}\mathbf{K}_{\mathbf{u}\boldsymbol{f}}\mathbf{K}_{\boldsymbol{f}\mathbf{u}}\right)\right] \\
&= -\frac{n}{2}\log 2\pi - \frac{1}{2}\log|\beta^{-1}\mathbf{I}_j| \\
&- \frac{1}{2}\left(\mathbf{y}_{:,j}^{\top}\beta\mathbf{I}_j\mathbf{y}_{:,j} - 2\mathbf{y}_{:,j}^{\top}\beta\mathbf{I}_j\mathbb{E}_{q(\mathbf{X}|\boldsymbol{\theta}_V)}\left[\mathbf{K}_{\boldsymbol{f}\mathbf{u}}\right]\mathbf{K}_{\mathbf{uu}}^{-1}\mathbf{u}_{:,j} + \mathbf{u}_{:,j}^{\top}\mathbf{K}_{\mathbf{uu}}^{-1}\beta\mathbb{E}_{q(\mathbf{X}|\boldsymbol{\theta}_V)}\left[\mathbf{K}_{\mathbf{u}\boldsymbol{f}}\mathbf{K}_{\boldsymbol{f}\mathbf{u}}\right]\mathbf{K}_{\mathbf{uu}}^{-1}\mathbf{u}_{:,j}\right) \\
&- \frac{\beta}{2}\mathrm{tr}\left(\mathbb{E}_{q(\mathbf{X}|\boldsymbol{\theta}_V)}\left[\mathbf{K}_{\boldsymbol{ff}}\right] - \mathbf{K}_{\mathbf{uu}}^{-1}\beta\mathbb{E}_{q(\mathbf{X}|\boldsymbol{\theta}_V)}\left[\mathbf{K}_{\mathbf{u}\boldsymbol{f}}\mathbf{K}_{\boldsymbol{f}\mathbf{u}}\right]\right) \\
&= -\frac{n}{2}\log 2\pi - \frac{1}{2}\log|\beta^{-1}\mathbf{I}_j| - \frac{1}{2}\left(\mathbf{y}_{:,j}^{\top}\beta\mathbf{I}_j\mathbf{y}_{:,j} - 2\mathbf{y}_{:,j}^{\top}\beta\mathbf{I}_j\boldsymbol{\psi}_1\mathbf{K}_{\mathbf{uu}}^{-1}\mathbf{u}_{:,j} + \mathbf{u}_{:,j}^{\top}\mathbf{K}_{\mathbf{uu}}^{-1}\beta\boldsymbol{\psi}_2\mathbf{K}_{\mathbf{uu}}^{-1}\mathbf{u}_{:,j}\right) \\
&- \frac{\beta}{2}\mathrm{tr}\left(\psi_0\right) + \frac{\beta}{2}\mathrm{tr}\left(\mathbf{K}_{\mathbf{uu}}^{-1}\boldsymbol{\psi}_2\right) \qquad \text{(D.14)}
\end{aligned}
$$

as in Section 5.5 the following are defined for notational convenience,

$$
\begin{aligned}
\psi_0 &\triangleq \mathbb{E}_{q(\mathbf{X}|\boldsymbol{\theta}_V)}\Big[\mathrm{tr}\left(\mathbf{K}_{\boldsymbol{ff}}\right)\Big] \\
\boldsymbol{\psi}_1 &\triangleq \mathbb{E}_{q(\mathbf{X}|\boldsymbol{\theta}_V)}\Big[\mathbf{K}_{\boldsymbol{f}\mathbf{u}}\Big] \\
\boldsymbol{\psi}_2 &\triangleq \mathbb{E}_{q(\mathbf{X}|\boldsymbol{\theta}_V)}\Big[\mathbf{K}_{\mathbf{u}\boldsymbol{f}}\mathbf{K}_{\boldsymbol{f}\mathbf{u}}\Big]
\end{aligned}
$$

D.2.2 Integrating Out $p(\mathbf{u})$

So far the bound obtained is still dependent on the inducing variables, $\mathbf{u}$. The final step will be to integrate over their prior uncertainty, $p(\mathbf{u})$. For prediction we also require access to $p(\mathbf{u}|\mathbf{y})$, or in practice an approximation to this posterior.

$$
\begin{aligned}
\log p(\mathbf{y}_{:,j}|\mathbf{Z}) &= \log \int \log p(\mathbf{y}_{:,j}|\mathbf{u}_{:,j})p(\mathbf{u}_{:,j}|\mathbf{Z})\, d\mathbf{u}_{:,j} \\
&\geq \log \int \log \hat{p}_j(\mathbf{y}_{:,j}|\mathbf{u}_{:,j})p(\mathbf{u}_{:,j}|\mathbf{Z})\, d\mathbf{u}_{:,j} - \mathrm{KL}\left(q(\mathbf{X}|\boldsymbol{\theta}_V)\,\|\, p(\mathbf{X})\right) \\
&= \log \hat{p}_j(\mathbf{y}_{:,j}|\mathbf{Z}) - \mathrm{KL}\left(q(\mathbf{X}|\boldsymbol{\theta}_V)\,\|\, p(\mathbf{X})\right)
\end{aligned}
$$

This integral does not require a further variational lower bound, which would further loosen the bound, and may instead be done analytically as follows.

$$
\begin{aligned}
&\log \int \Big(\exp \log \hat{p}_j(\mathbf{y}_{:,j}|\mathbf{u}_{:,j})\Big) p(\mathbf{u}_{:,j}|\mathbf{Z})\, d\mathbf{u}_{:,j} \\
&\quad = \log \int \Bigg(\exp\Bigg[-\frac{n}{2}\log 2\pi - \frac{1}{2}\log|\beta^{-1}\mathbf{I}_j| - \frac{1}{2}\mathrm{tr}\left(\mathbf{y}_{:,j}^{\top}\beta\mathbf{I}_j\mathbf{y}_{:,j} - 2\mathbf{y}_{:,j}^{\top}\beta\mathbf{I}_j\boldsymbol{\psi}_1\mathbf{K}_{\mathbf{uu}}^{-1}\mathbf{u}_{:,j}\right. \\
&\qquad\quad \left. + \mathbf{u}_{:,j}^{\top}\mathbf{K}_{\mathbf{uu}}^{-1}\beta\boldsymbol{\psi}_2\mathbf{K}_{\mathbf{uu}}^{-1}\mathbf{u}_{:,j}\right) \\
&\quad - \frac{\beta}{2}\mathrm{tr}\left(\psi_0\right) + \frac{\beta}{2}\mathrm{tr}\left(\mathbf{K}_{\mathbf{uu}}^{-1}\boldsymbol{\psi}_2\right)\Bigg]\Bigg) p(\mathbf{u}_{:,j}|\mathbf{Z})\, d\mathbf{u}_{:,j} \\
&\quad = \log\Bigg\{\exp\Bigg[-\frac{n}{2}\log 2\pi - \frac{1}{2}\log|\beta^{-1}\mathbf{I}_j| - \frac{\beta}{2}\mathrm{tr}\left(\psi_0\right) + \frac{\beta}{2}\mathrm{tr}\left(\mathbf{K}_{\mathbf{uu}}^{-1}\boldsymbol{\psi}_2\right) - \frac{1}{2}\mathrm{tr}\left(\mathbf{y}_{:,j}^{\top}\beta\mathbf{I}_j\mathbf{y}_{:,j}\right)\Bigg] \cdot \\
&\quad \Bigg[\int \exp\Big(-2\mathbf{y}_{:,j}^{\top}\beta\mathbf{I}_j\boldsymbol{\psi}_1\mathbf{K}_{\mathbf{uu}}^{-1}\mathbf{u}_{:,j} + \mathbf{u}_{:,j}^{\top}\mathbf{K}_{\mathbf{uu}}^{-1}\beta\boldsymbol{\psi}_2\mathbf{K}_{\mathbf{uu}}^{-1}\mathbf{u}_{:,j})\Big) p(\mathbf{u}_{:,j}|\mathbf{Z})\, d\mathbf{u}_{:,j}\Bigg]\Bigg\} \\
&\quad = -\frac{n}{2}\log 2\pi - \frac{1}{2}\log|\beta^{-1}\mathbf{I}_j| - \frac{\beta}{2}\mathrm{tr}\left(\psi_0\right) + \frac{\beta}{2}\mathrm{tr}\left(\mathbf{K}_{\mathbf{uu}}^{-1}\boldsymbol{\psi}_2\right) - \frac{1}{2}\mathrm{tr}\left(\mathbf{y}_{:,j}^{\top}\beta\mathbf{I}_j\mathbf{y}_{:,j}\right) \\
&\quad + \log \int \exp\Big(-\frac{1}{2}\mathrm{tr}\left(-2\mathbf{y}^{\top}\beta\mathbf{I}_j\boldsymbol{\psi}_1\mathbf{K}_{\mathbf{uu}}^{-1}\mathbf{u}_{:,j} + \mathbf{u}_{:,j}^{\top}\mathbf{K}_{\mathbf{uu}}^{-1}\beta\boldsymbol{\psi}_2\mathbf{K}_{\mathbf{uu}}^{-1}\mathbf{u}_{:,j}\right) \\
&\qquad - \frac{m}{2}\log 2\pi - \frac{1}{2}\log|\mathbf{K}_{\mathbf{uu}}| - \frac{1}{2}\mathrm{tr}\left(\mathbf{u}_{:,j}^{\top}\mathbf{K}_{\mathbf{uu}}^{-1}\mathbf{u}_{:,j}\right)\Big)\, d\mathbf{u}_{:,j} \qquad \text{(substitute } p(\mathbf{u}_{:,j}|\mathbf{Z})\text{)} \\
&\quad = -\frac{n}{2}\log 2\pi - \frac{1}{2}\log|\beta^{-1}\mathbf{I}_j| - \frac{\beta}{2}\mathrm{tr}\left(\psi_0\right) + \frac{\beta}{2}\mathrm{tr}\left(\mathbf{K}_{\mathbf{uu}}^{-1}\boldsymbol{\psi}_2\right) - \frac{1}{2}\mathrm{tr}\left(\mathbf{y}_{:,j}^{\top}\beta\mathbf{I}_j\mathbf{y}_{:,j}\right) \\
&\quad + \log \int \exp\Big(\frac{1}{2}\mathrm{tr}\left(-2\mathbf{y}_{:,j}^{\top}\beta\mathbf{I}_j\boldsymbol{\psi}_1\mathbf{K}_{\mathbf{uu}}^{-1}\mathbf{u}_{:,j} + \mathbf{u}_{:,j}^{\top}(\mathbf{K}_{\mathbf{uu}}^{-1}\beta\boldsymbol{\psi}_2\mathbf{K}_{\mathbf{uu}}^{-1} + \mathbf{K}_{\mathbf{uu}}^{-1})\mathbf{u}_{:,j}\right) \\
&\qquad - \frac{m}{2}\log 2\pi - \frac{1}{2}\log|\mathbf{K}_{\mathbf{uu}}|\Big)\, d\mathbf{u}_{:,j} \qquad \text{(Complete for } \mathbf{u}_{:,j}\text{)}
\end{aligned}
$$

To complete the square for $\mathbf{u}$, we must introduce extra terms such that we have the following form, with some additional constant terms in $\mathbf{u}$.

$$
\log \mathcal{N}\left(\mathbf{u}|\mathbf{m},\mathbf{R}\right) = -\frac{n}{2}\log 2\pi - \frac{1}{2}\log|\mathbf{R}| - \frac{1}{2}\mathrm{tr}\left(\mathbf{m}^{\top}\mathbf{R}^{-1}\mathbf{m} + 2\mathbf{u}^{\top}\mathbf{R}^{-1}\mathbf{m} - \mathbf{u}^{\top}\mathbf{R}^{-1}\mathbf{u}\right). \tag{D.15}
$$

In this case the Gaussian $\mathcal{N}\left(\mathbf{u}|\mathbf{m}_{:,j},\mathbf{R}\right)$ plus other terms may be found as follows,

$$
\frac{1}{2}\mathrm{tr}\left(-2\mathbf{y}_{:,j}^{\top}\beta\mathbf{I}_j\boldsymbol{\psi}_1\mathbf{K}_{\mathbf{uu}}^{-1}\mathbf{u}_{:,j} + \mathbf{u}_{:,j}^{\top}\underbrace{\left(\mathbf{K}_{\mathbf{uu}}^{-1}\beta\boldsymbol{\psi}_2\mathbf{K}_{\mathbf{uu}}^{-1} + \mathbf{K}_{\mathbf{uu}}^{-1}\right)}_{\mathbf{R}^{-1}}\mathbf{u}_{:,j}\right) - \frac{m}{2}\log 2\pi - \frac{1}{2}\log|\mathbf{K}_{\mathbf{uu}}|
$$

And so by matching linear terms in $\mathbf{u}_{:,j}$, $\mathbf{u}_{:,j}^{\top}\mathbf{R}^{-1}\mathbf{m}_{:,j} = \mathbf{u}_{:,j}^{\top}\mathbf{K}_{\mathbf{uu}}^{-1}\boldsymbol{\psi}_1{}^{\top}\beta\mathbf{y}_{:,j}$, and quadratic terms

$$\frac{1}{2}\mathrm{tr}\left(-2\underbrace{\mathbf{y}_{:,j}^{\top}\beta\mathbf{I}_j\boldsymbol{\psi}_1\mathbf{K}_{\mathbf{uu}}^{-1}}_{\mathbf{m}_{:,j}^{\top}\mathbf{R}^{-1}}\mathbf{u}_{:,j} + \mathbf{u}_{:,j}^{\top}\underbrace{(\mathbf{K}_{\mathbf{uu}}^{-1}\beta\boldsymbol{\psi}_2\mathbf{K}_{\mathbf{uu}}^{-1} + \mathbf{K}_{\mathbf{uu}}^{-1})}_{\mathbf{R}^{-1}}\mathbf{u}_{:,j}\right) - \frac{m}{2}\log 2\pi - \frac{1}{2}\log|\mathbf{K}_{\mathbf{uu}}|$$

$$= \log\mathcal{N}\left(\mathbf{u}_{:,j}|\mathbf{m}_{:,j}, \mathbf{R}^{-1}\right) + \underbrace{\frac{1}{2}\mathrm{tr}\left(\mathbf{m}_{:,j}^{\top}\mathbf{R}^{-1}\mathbf{m}_{:,j}\right) + \frac{1}{2}\log|\mathbf{R}| - \frac{1}{2}\log|\mathbf{K}_{\mathbf{uu}}^{-1}|}_{G}$$

So in this case, and so the mean and covariance are,

$$q(\mathbf{u}|\boldsymbol{\theta}_V) = \mathcal{N}\left(\mathbf{u}_{:,j}|\mathbf{m}_{:,j}, \mathbf{R}\right) \tag{D.16}$$

$$\mathbf{m}_{:,j} = \mathbf{R}\mathbf{K}_{\mathbf{uu}}^{-1}\boldsymbol{\psi}_1{}^{\top}\beta\mathbf{y}_{:,j} \tag{D.17}$$

$$\mathbf{R} = (\mathbf{K}_{\mathbf{uu}}^{-1}\beta\boldsymbol{\psi}_2\mathbf{K}_{\mathbf{uu}}^{-1} + \mathbf{K}_{\mathbf{uu}}^{-1})^{-1}. \tag{D.18}$$

Note that these match the optimal distribution for $q(\mathbf{u}|\boldsymbol{\theta}_V)$ given in the sparse GP Section 3.3, i.e. $p(\mathbf{u}_{:,j}|\mathbf{y}_{:,j}) \approx \mathcal{N}\left(\mathbf{u}_{:,j}|\mathbf{m}_{:,j}, \mathbf{R}\right) \triangleq q(\mathbf{u}|\boldsymbol{\theta}_V)$, however in that case $\boldsymbol{\psi}_1$ and $\boldsymbol{\psi}_2$ are substituted for $\mathbf{K}_{\boldsymbol{f}\mathbf{u}}$ and $\mathbf{K}_{\mathbf{u}\boldsymbol{f}}\mathbf{K}_{\boldsymbol{f}\mathbf{u}}$ respectively, as they have not had their expectation taken with respect to $q(\mathbf{X}|\boldsymbol{\theta}_V)$. This result arises as we have calculated an approximation to $p(\mathbf{y}_{:,j}|\mathbf{u}_{:,j})p(\mathbf{u}_{:,j}) = p(\mathbf{u}_{:,j}|\mathbf{y}_{:,j})$, using the approximate $\hat{p}_j(\mathbf{y}_{:,j}|\mathbf{u}_{:,j})$.

The additional terms, G, arise from terms that we were missing to complete the square, $-\frac{1}{2}\mathrm{tr}\left(\mathbf{m}_{:,j}^{\top}\mathbf{R}^{-1}\mathbf{m}_{:,j}\right)$ and $-\frac{1}{2}\log|\mathbf{R}|$, and terms that were left over, $-\frac{1}{2}\log|\mathbf{K}_{\mathbf{uu}}^{-1}|$.

By plugging the result back into our marginalisation of $\mathbf{u}_{:,j}$ and using the fact that a probability distribution integrates to 1, $\int\mathcal{N}\left(\mathbf{u}_{:,j}|\mathbf{m}_{:,j}, \mathbf{R}\right)d\mathbf{u}_{:,j} = 1$,

$$\begin{aligned}
&\int \exp(\log\mathcal{N}\left(\mathbf{u}|\mathbf{m}, \mathbf{R}\right) + G)\, d\mathbf{u} \\
&= \int \mathcal{N}\left(\mathbf{u}|\mathbf{m}, \mathbf{R}\right)\exp(G)\, d\mathbf{u} \\
&= \exp(G)\int \mathcal{N}\left(\mathbf{u}|\mathbf{m}, \mathbf{R}\right)\, d\mathbf{u} \\
&= \exp(G),
\end{aligned} \tag{D.19}$$

Where G are terms independent of the distribution for $\mathbf{u}$; the integral becomes,

$$\begin{aligned}
&\log \int \Big(\exp \log \hat{p}_j(\mathbf{y}_{:,j}|\mathbf{u}_{:,j})\Big) p(\mathbf{u}_{:,j}|\mathbf{Z})\, d\mathbf{u}_{:,j} \\
&\quad = -\frac{n}{2}\log 2\pi - \frac{1}{2}\log|\beta^{-1}\mathbf{I}_j| - \frac{\beta}{2}\mathrm{tr}\left(\psi_0\right) + \frac{\beta}{2}\mathrm{tr}\left(\mathbf{K}_{\mathbf{uu}}^{-1}\boldsymbol{\psi}_2\right) - \frac{1}{2}\mathrm{tr}\left(\mathbf{y}_{:,j}^{\top}\beta\mathbf{I}_j\mathbf{y}_{:,j}\right) \\
&\quad + \log \int \exp\Big(\log \mathcal{N}\left(\mathbf{u}_{:,j}|\mathbf{m}_{:,j}, \mathbf{R}^{-1}\right) + \frac{1}{2}\mathrm{tr}\left(\mathbf{m}_{:,j}^{\top}\mathbf{R}^{-1}\mathbf{m}_{:,j}\right) + \frac{1}{2}\log|\mathbf{R}| - \frac{1}{2}\log|\mathbf{K}_{\mathbf{uu}}^{-1}|\Big)\, d\mathbf{u}_{:,j} \\
&\quad = -\frac{n}{2}\log 2\pi - \frac{1}{2}\log|\beta^{-1}\mathbf{I}_j| - \frac{\beta}{2}\mathrm{tr}\left(\psi_0\right) + \frac{\beta}{2}\mathrm{tr}\left(\mathbf{K}_{\mathbf{uu}}^{-1}\boldsymbol{\psi}_2\right) - \frac{1}{2}\mathrm{tr}\left(\mathbf{y}_{:,j}^{\top}\beta\mathbf{I}_j\mathbf{y}_{:,j}\right) \\
&\quad + \frac{1}{2}\mathrm{tr}\left(\mathbf{m}_{:,j}^{\top}\mathbf{R}^{-1}\mathbf{m}_{:,j}\right) + \frac{1}{2}\log|\mathbf{R}| - \frac{1}{2}\log|\mathbf{K}_{\mathbf{uu}}^{-1}|
\end{aligned}$$

where the final line uses Equation (D.19).

The bound on the marginal likelihood then becomes,

$$\begin{aligned}
\log p(\mathbf{y}_{:,j}|\mathbf{Z}) &\geq -\mathrm{KL}\left(q(\mathbf{X}|\boldsymbol{\theta}_V)\,\|\,p(\mathbf{X})\right) + \log \int \left(\exp\log \hat{p}_j(\mathbf{y}_{:,j}|\mathbf{u}_{:,j})\right) p(\mathbf{u}_{:,j}|\mathbf{Z})\, d\mathbf{u}_{:,j} \\
&= -\mathrm{KL}\left(q(\mathbf{X}|\boldsymbol{\theta}_V)\,\|\,p(\mathbf{X})\right) - \frac{n}{2}\log 2\pi - \frac{1}{2}\log|\beta^{-1}\mathbf{I}_j| - \frac{\beta}{2}\mathrm{tr}\left(\psi_0 - \mathbf{K}_{\mathbf{uu}}^{-1}\boldsymbol{\psi}_2\right) \\
&- \frac{1}{2}\mathrm{tr}\left(\mathbf{y}_{:,j}^{\top}\beta\mathbf{I}_j\mathbf{y}_{:,j}\right) + \frac{1}{2}\mathrm{tr}\left(\mathbf{m}_{:,j}^{\top}\mathbf{R}^{-1}\mathbf{m}_{:,j}\right) + \frac{1}{2}\log|\mathbf{R}| - \frac{1}{2}\log|\mathbf{K}_{\mathbf{uu}}^{-1}| \\
&= \log \hat{p}_j(\mathbf{y}_{:,j}|\mathbf{Z}) - \mathrm{KL}\left(q(\mathbf{X}|\boldsymbol{\theta}_V)\,\|\,p(\mathbf{X})\right)
\end{aligned}$$

By substituting $\mathbf{R}$ and $\mathbf{m}_{:,j}$,

$$\begin{aligned}
\log p(\mathbf{y}_{:,j}|\mathbf{Z}) = &-\frac{n}{2}\log 2\pi - \frac{1}{2}\log|\beta^{-1}\mathbf{I}| - \frac{\beta}{2}\mathrm{tr}\left(\psi_0\right) + \frac{\beta}{2}\mathrm{tr}\left(\mathbf{K}_{\mathbf{uu}}^{-1}\boldsymbol{\psi}_2\right) - \frac{1}{2}\mathrm{tr}\left(\mathbf{y}_{:,j}^\top\beta^{-1}\mathbf{I}\mathbf{y}_{:,j}\right)\\
&+ \frac{1}{2}\mathrm{tr}\left(\mathbf{y}_{:,j}^\top\beta\boldsymbol{\psi}_1\mathbf{K}_{\mathbf{uu}}^{-1}\cancel{\mathbf{R}\mathbf{R}^{-1}}\mathbf{R}\mathbf{K}_{\mathbf{uu}}^{-1}\boldsymbol{\psi}_1{}^\top\beta\mathbf{y}_{:,j}\right)\\
&+ \frac{1}{2}\log|(\mathbf{K}_{\mathbf{uu}}^{-1}\beta\boldsymbol{\psi}_2\mathbf{K}_{\mathbf{uu}}^{-1} + \mathbf{K}_{\mathbf{uu}}^{-1})^{-1}| - \frac{1}{2}\log|\mathbf{K}_{\mathbf{uu}}^{-1}|\\
= &-\frac{n}{2}\log 2\pi - \frac{1}{2}\log|\beta^{-1}\mathbf{I}| - \frac{\beta}{2}\mathrm{tr}\left(\psi_0\right) + \frac{\beta}{2}\mathrm{tr}\left(\mathbf{K}_{\mathbf{uu}}^{-1}\boldsymbol{\psi}_2\right)\\
&- \frac{1}{2}\mathrm{tr}\left(\mathbf{y}_{:,j}^\top(\beta^{-1}\mathbf{I} - \beta\boldsymbol{\psi}_1\mathbf{K}_{\mathbf{uu}}^{-1}(\mathbf{K}_{\mathbf{uu}}^{-1}\beta\boldsymbol{\psi}_2\mathbf{K}_{\mathbf{uu}}^{-1} + \mathbf{K}_{\mathbf{uu}}^{-1})^{-1}\mathbf{K}_{\mathbf{uu}}^{-1}\boldsymbol{\psi}_1{}^\top\beta)\mathbf{y}_{:,j}\right)\\
&+ \frac{1}{2}\log|(\mathbf{K}_{\mathbf{uu}}^{-1}\beta\boldsymbol{\psi}_2\mathbf{K}_{\mathbf{uu}}^{-1} + \mathbf{K}_{\mathbf{uu}}^{-1})^{-1}| - \frac{1}{2}\log|\mathbf{K}_{\mathbf{uu}}^{-1}| \qquad \text{(D.20)}\\
= &-\frac{n}{2}\log 2\pi - \frac{1}{2}\log|\beta^{-1}\mathbf{I}| - \frac{\beta}{2}\mathrm{tr}\left(\psi_0\right) + \frac{\beta}{2}\mathrm{tr}\left(\mathbf{K}_{\mathbf{uu}}^{-1}\boldsymbol{\psi}_2\right)\\
&- \frac{1}{2}\mathrm{tr}\left(\mathbf{y}_{:,j}^\top(\beta^{-1}\mathbf{I} - \beta\boldsymbol{\psi}_1(\beta\boldsymbol{\psi}_2 + \mathbf{K}_{\mathbf{uu}})^{-1}\boldsymbol{\psi}_1{}^\top\beta)\mathbf{y}_{:,j}\right)\\
&+ \frac{1}{2}\log|\mathbf{K}_{\mathbf{uu}}^{-1}(\beta\boldsymbol{\psi}_2 + \mathbf{K}_{\mathbf{uu}})^{-1}\mathbf{K}_{\mathbf{uu}}^{-1}| - \frac{1}{2}\log|\mathbf{K}_{\mathbf{uu}}^{-1}|\\
= &-\frac{n}{2}\log 2\pi - \frac{1}{2}\log|\beta^{-1}\mathbf{I}| - \frac{\beta}{2}\mathrm{tr}\left(\psi_0\right) + \frac{\beta}{2}\mathrm{tr}\left(\mathbf{K}_{\mathbf{uu}}^{-1}\boldsymbol{\psi}_2\right)\\
&- \frac{1}{2}\mathrm{tr}\left(\mathbf{y}_{:,j}^\top(\beta^{-1}\mathbf{I} - \beta\boldsymbol{\psi}_1(\beta\boldsymbol{\psi}_2 + \mathbf{K}_{\mathbf{uu}})^{-1}\boldsymbol{\psi}_1{}^\top\beta)\mathbf{y}_{:,j}\right)\\
&- \frac{1}{2}\log|(\beta\boldsymbol{\psi}_2 + \mathbf{K}_{\mathbf{uu}})| + \frac{1}{2}\log|\mathbf{K}_{\mathbf{uu}}^{-1}|\cancel{+\frac{1}{2}\log|\mathbf{K}_{\mathbf{uu}}^{-1}| - \frac{1}{2}\log|\mathbf{K}_{\mathbf{uu}}^{-1}|}
\end{aligned}$$

where the third line uses the equality,

$$\mathbf{K}_{\mathbf{uu}}^{-1}(\mathbf{K}_{\mathbf{uu}}^{-1}\beta\boldsymbol{\psi}_2 + \mathbf{K}_{\mathbf{uu}}^{-1})^{-1}\mathbf{K}_{\mathbf{uu}}^{-1} = \mathbf{K}_{\mathbf{uu}}(\beta\boldsymbol{\psi}_2 + \mathbf{K}_{\mathbf{uu}})^{1}\mathbf{K}_{\mathbf{uu}} \qquad \text{(D.21)}$$

Finally this can be rewritten, including the sum over dimensions

$$\begin{aligned}
\log p(\mathbf{Y}|\mathbf{Z}) \geq \sum_{j=1}^{p} &-\log\left(2\pi\right)^{\frac{n}{2}} + \log|\beta^{-1}\mathbf{I}|^{-\frac{1}{2}} - \frac{1}{2}\log|(\beta\boldsymbol{\psi}_2 + \mathbf{K}_{\mathbf{uu}})| + \log|\mathbf{K}_{\mathbf{uu}}^{-1}|^{\frac{1}{2}} \qquad \text{(D.22)}\\
&- \frac{1}{2}\mathrm{tr}\left(\mathbf{y}_{:,j}^\top(\beta^{-1}\mathbf{I} - \beta\boldsymbol{\psi}_1(\beta\boldsymbol{\psi}_2 + \mathbf{K}_{\mathbf{uu}})^{-1}\boldsymbol{\psi}_1{}^\top\beta)\mathbf{y}_{:,j}\right) - \frac{\beta}{2}\mathrm{tr}\left(\psi_0\right) + \frac{\beta}{2}\mathrm{tr}\left(\mathbf{K}_{\mathbf{uu}}^{-1}\boldsymbol{\psi}_2\right)\\
= &\log\left[\frac{\beta^{\frac{n}{2}}|\mathbf{K}_{\mathbf{uu}}^{-1}|^{\frac{1}{2}}e^{-\frac{1}{2}\mathrm{tr}\left(\mathbf{y}_{:,j}^\top\boldsymbol{D}\mathbf{y}_{:,j}\right)}}{2\pi^{\frac{n}{2}}|(\beta\boldsymbol{\psi}_2 + \mathbf{K}_{\mathbf{uu}})|^{\frac{1}{2}}}\right] - \frac{\beta}{2}\mathrm{tr}\left(\psi_0\right) + \frac{\beta}{2}\mathrm{tr}\left(\mathbf{K}_{\mathbf{uu}}^{-1}\boldsymbol{\psi}_2\right) \qquad \text{(D.23)}
\end{aligned}$$

$\boldsymbol{D} = \beta^{-1}\mathbf{I} - \beta\boldsymbol{\psi}_1(\beta\boldsymbol{\psi}_2 + \mathbf{K}_{\mathbf{uu}})^{-1}{\boldsymbol{\psi}_1}^{\top}\beta$, which exactly matches the form given in the original publication (Titsias, 2009); this is the result given in the BGPLVM Section 5.5.

D.2.3 Completing In $\mathbf{y}$

The form given in the original publication, as shown in equation (D.23) does not make it clear that the bound is Gaussian in $\mathbf{y}_{:,j}$, though it does have a beneficial computational complexity of $\mathcal{O}(nm^2)$. As is shown in Section 5.6, it is not only insightful, Section 5.6.2, see the form of the bound in the form of a Gaussian with regularising terms, it also allows for the applicability of the Laplace approximation with no modification. Section 5.6 shows that this can be used to extend the BGPLVM model to non-Gaussian likelihoods, a model called the LABGPLVM, as well as its missing data counterpart, Section 5.6.4.

In the following section, a more verbose derivation of the equations given in the Section 5.6.2 is provided, providing additional insights to the BGPLVM model and paving the way for a subsequent Laplace approximation.

The first step is to complete the square with respect to $\mathbf{y}$, starting from the form of the bound given in Equation (D.20), and substitute in $\mathbf{R}$ in order to simplify.

$$
\begin{aligned}
&\log \hat{p}_j(\mathbf{y}_{:,j}|\mathbf{Z}) = \\
&= -\frac{n}{2}\log 2\pi - \frac{1}{2}\log|\beta^{-1}\mathbf{I}_j| - \frac{\beta}{2}\mathrm{tr}\left(\psi_0\right) + \frac{\beta}{2}\mathrm{tr}\left(\mathbf{K}_{\mathbf{uu}}^{-1}\boldsymbol{\psi}_2\right) + \frac{1}{2}\log|\mathbf{R}| \\
&- \frac{1}{2}\mathrm{tr}\left(\mathbf{y}_{:,j}^{\top}(\beta\mathbf{I}_j - \beta^{\top}\boldsymbol{\psi}_1\mathbf{K}_{\mathbf{uu}}^{-1}(\mathbf{K}_{\mathbf{uu}}^{-1}\beta\boldsymbol{\psi}_2\mathbf{K}_{\mathbf{uu}}^{-1} + \mathbf{K}_{\mathbf{uu}}^{-1})^{-1}\mathbf{K}_{\mathbf{uu}}^{-1}{\boldsymbol{\psi}_1}^{\top}\beta)\mathbf{y}_{:,j}\right) - \frac{1}{2}\log|\mathbf{K}_{\mathbf{uu}}^{-1}| \\
&= -\frac{n}{2}\log 2\pi - \frac{1}{2}\log|\beta^{-1}\mathbf{I}_j| - \frac{\beta}{2}\mathrm{tr}\left(\psi_0\right) + \frac{\beta}{2}\mathrm{tr}\left(\mathbf{K}_{\mathbf{uu}}^{-1}\boldsymbol{\psi}_2\right) + \frac{1}{2}\log|(\mathbf{K}_{\mathbf{uu}}^{-1}\beta\boldsymbol{\psi}_2\mathbf{K}_{\mathbf{uu}}^{-1} + \mathbf{K}_{\mathbf{uu}}^{-1})^{-1}| \\
&- \frac{1}{2}\mathrm{tr}\left(\mathbf{y}_{:,j}^{\top}\underbrace{(\beta\mathbf{I}_j - \beta^{\top}\boldsymbol{\psi}_1(\beta\boldsymbol{\psi}_2 + \mathbf{K}_{\mathbf{uu}})^{-1}{\boldsymbol{\psi}_1}^{\top}\beta)}_{\hat{\mathbf{K}}^{-1}}\mathbf{y}_{:,j}\right) - \frac{1}{2}\log|\mathbf{K}_{\mathbf{uu}}^{-1}| \\
&= -\frac{1}{2}\log|\beta^{-1}\mathbf{I}_j| - \frac{\beta}{2}\mathrm{tr}\left(\psi_0\right) + \frac{\beta}{2}\mathrm{tr}\left(\mathbf{K}_{\mathbf{uu}}^{-1}\boldsymbol{\psi}_2\right) + \log\mathcal{N}\left(\mathbf{y}_{:,j}|\mathbf{0},\hat{\mathbf{K}}\right) \\
&+ \frac{1}{2}\log|(\mathbf{K}_{\mathbf{uu}}^{-1}\beta\boldsymbol{\psi}_2\mathbf{K}_{\mathbf{uu}}^{-1} + \mathbf{K}_{\mathbf{uu}}^{-1})^{-1}| - \frac{1}{2}\log|\mathbf{K}_{\mathbf{uu}}^{-1}| \underbrace{+\frac{1}{2}\log|\hat{\mathbf{K}}|}_{\text{required to complete the square}} \\
&= -\frac{\beta}{2}\mathrm{tr}\left(\psi_0\right) + \frac{\beta}{2}\mathrm{tr}\left(\mathbf{K}_{\mathbf{uu}}^{-1}\boldsymbol{\psi}_2\right) + \log\mathcal{N}\left(\mathbf{y}_{:,j}|\mathbf{0},\hat{\mathbf{K}}\right) + \frac{1}{2}\log|(\mathbf{K}_{\mathbf{uu}}^{-1}\beta\boldsymbol{\psi}_2\mathbf{K}_{\mathbf{uu}}^{-1} + \mathbf{K}_{\mathbf{uu}}^{-1})^{-1}| \\
&- \frac{1}{2}\log|\mathbf{K}_{\mathbf{uu}}^{-1}| + \frac{1}{2}\log|\hat{\mathbf{K}}| - \frac{1}{2}\log|\beta^{-1}\mathbf{I}_j|.
\end{aligned}
$$

As in Section 5.5.1 this form be simplified using the matrix inversion lemma (Section A.2.1,

$$
\begin{aligned}
\hat{\mathbf{K}} &= \left(\beta\mathbf{I}_j - \beta^\top \boldsymbol{\psi}_1(\beta\boldsymbol{\psi}_2 + \mathbf{K}_{\mathbf{uu}})^{-1}{\boldsymbol{\psi}_1}^\top\beta\right)^{-1} \\
&= \left(\underbrace{\beta\mathbf{I}_j}_{A} + \underbrace{-\beta\mathbf{I}_j^\top\boldsymbol{\psi}_1}_{U}\underbrace{(\beta\boldsymbol{\psi}_2 + \mathbf{K}_{\mathbf{uu}})^{-1}}_{B}\underbrace{{\boldsymbol{\psi}_1}^\top\beta\mathbf{I}_j}_{V}\right)^{-1} \\
&= \beta^{-1}\mathbf{I}_j \\
&- (\beta^{-1}\mathbf{I}_j)(-\beta\mathbf{I}_j\boldsymbol{\psi}_1)\left((\beta\boldsymbol{\psi}_2 + \mathbf{K}_{\mathbf{uu}}) + ({\boldsymbol{\psi}_1}^\top\beta\mathbf{I}_j)(\beta^{-1}\mathbf{I}_j)(-\beta\mathbf{I}_j\boldsymbol{\psi}_1)\right)^{-1}({\boldsymbol{\psi}_1}^\top\beta\mathbf{I}_j)(\beta^{-1}\mathbf{I}_j) \\
&= \beta^{-1}\mathbf{I}_j + \boldsymbol{\psi}_1\left((\beta\boldsymbol{\psi}_2 + \mathbf{K}_{\mathbf{uu}}) - ({\boldsymbol{\psi}_1}^\top\beta\boldsymbol{\psi}_1)\right)^{-1}{\boldsymbol{\psi}_1}^\top \\
&= \beta^{-1}\mathbf{I}_j + \boldsymbol{\psi}_1\left(\mathbf{K}_{\mathbf{uu}} + \beta(\boldsymbol{\psi}_2 - {\boldsymbol{\psi}_1}^\top\boldsymbol{\psi}_1)\right)^{-1}{\boldsymbol{\psi}_1}^\top
\end{aligned}
$$

This is very similar to the sparse GP bound, where $\mathbf{K}_{\mathbf{uu}}$ has been replaced by $\mathbf{K}_{\mathbf{uu}} + \beta(\boldsymbol{\psi}_2 - {\boldsymbol{\psi}_1}^\top\boldsymbol{\psi}_1)$. See Section 5.5.1 for more details on this interesting interpretation.

The determinants may be collated using the matrix determinant lemma, Section A.2.2,

$$
\begin{aligned}
&\frac{1}{2}\log|(\mathbf{K}_{\mathbf{uu}}^{-1}\beta\boldsymbol{\psi}_2\mathbf{K}_{\mathbf{uu}}^{-1} + \mathbf{K}_{\mathbf{uu}}^{-1})^{-1}| - \frac{1}{2}\log|\mathbf{K}_{\mathbf{uu}}^{-1}| + \frac{1}{2}\log|\hat{\mathbf{K}}| - \frac{1}{2}\log|\beta^{-1}\mathbf{I}_j| \\
&= \frac{1}{2}\log|\underbrace{\beta^{-1}\mathbf{I}_j}_{A} + \underbrace{\boldsymbol{\psi}_1}_{U}\underbrace{\left(\mathbf{K}_{\mathbf{uu}} + \beta(\boldsymbol{\psi}_2 - {\boldsymbol{\psi}_1}^\top\boldsymbol{\psi}_1)\right)^{-1}}_{W}\underbrace{{\boldsymbol{\psi}_1}^\top}_{V^\top}| \\
&- \frac{1}{2}\log|\beta^{-1}\mathbf{I}_j| + \frac{1}{2}\log|(\mathbf{K}_{\mathbf{uu}}^{-1}\beta\boldsymbol{\psi}_2\mathbf{K}_{\mathbf{uu}}^{-1} + \mathbf{K}_{\mathbf{uu}}^{-1})^{-1}| - \frac{1}{2}\log|\mathbf{K}_{\mathbf{uu}}^{-1}| \\
&= \frac{1}{2}\log|\mathbf{K}_{\mathbf{uu}} + \beta(\boldsymbol{\psi}_2 - {\boldsymbol{\psi}_1}^\top\boldsymbol{\psi}_1) + {\boldsymbol{\psi}_1}^\top\beta\mathbf{I}_j\boldsymbol{\psi}_1||(\mathbf{K}_{\mathbf{uu}} + \beta(\boldsymbol{\psi}_2 - {\boldsymbol{\psi}_1}^\top\boldsymbol{\psi}_1))^{-1}||\beta^{-1}\mathbf{I}_j| \\
&- \frac{1}{2}\log|\beta^{-1}\mathbf{I}_j| + \frac{1}{2}\log|(\mathbf{K}_{\mathbf{uu}}^{-1}\beta\boldsymbol{\psi}_2\mathbf{K}_{\mathbf{uu}}^{-1} + \mathbf{K}_{\mathbf{uu}}^{-1})^{-1}| - \frac{1}{2}\log|\mathbf{K}_{\mathbf{uu}}^{-1}| \\
&= +\frac{1}{2}\log|\left(\mathbf{K}_{\mathbf{uu}} + \beta(\boldsymbol{\psi}_2 - {\boldsymbol{\psi}_1}^\top\boldsymbol{\psi}_1)\right)^{-1}| + \frac{1}{2}\log|\mathbf{K}_{\mathbf{uu}}|
\end{aligned}
$$

where the final line arises as a result of many cancellations.

The final bound, completed with respect to $\mathbf{y}_{:,j}$ is then given by:

$$
\begin{aligned}
&\log p(\mathbf{y}_{:,j}|\mathbf{Z}) = \\
&\geq \log \mathcal{N}\left(\mathbf{y}_{:,j}|\mathbf{0}, \hat{\mathbf{K}}\right) + \frac{1}{2}\log\left|(\mathbf{K}_{\mathbf{uu}} + \beta(\boldsymbol{\psi}_2 - \boldsymbol{\psi}_1{}^{\top}\boldsymbol{\psi}_1))^{-1}\right| + \frac{1}{2}\log|\mathbf{K}_{\mathbf{uu}}| \\
&- \frac{\beta}{2}\mathrm{tr}\left(\psi_0\right) + \frac{\beta}{2}\mathrm{tr}\left(\mathbf{K}_{\mathbf{uu}}^{-1}\boldsymbol{\psi}_2\right) - \mathrm{KL}\left(q(\mathbf{X}|\boldsymbol{\theta}_V) \,\|\, p(\mathbf{X})\right) \\
&= \log \hat{p}_j(\mathbf{y}_{:,j}|\mathbf{Z}) - \frac{\beta}{2}\mathrm{tr}\left(\psi_0\right) + \frac{\beta}{2}\mathrm{tr}\left(\mathbf{K}_{\mathbf{uu}}^{-1}\boldsymbol{\psi}_2\right) - \mathrm{KL}\left(q(\mathbf{X}|\boldsymbol{\theta}_V) \,\|\, p(\mathbf{X})\right)
\end{aligned}
$$

So far the bound with respect to a single dimension has been considered, we can take the product of each dimension to provide the final simplified bound for the BGPLVM,

$$
\begin{aligned}
\log p(\mathbf{y}|\mathbf{Z}) &= \log \prod_{j=1}^{p} p(\mathbf{y}_j|\mathbf{Z}) \\
&= \sum_{j=1}^{p} \log p(\mathbf{y}_j|\mathbf{Z}) \\
&\geq \left[\sum_{j=1}^{p} \log \hat{p}(\mathbf{y}_j|\mathbf{Z}) - \frac{\beta}{2}\mathrm{tr}\left(\psi_0\right) + \frac{\beta}{2}\mathrm{tr}\left(\mathbf{K}_{\mathbf{uu}}^{-1}\boldsymbol{\psi}_2\right)\right] - \mathrm{KL}\left(q(\mathbf{X}|\boldsymbol{\theta}_V) \,\|\, p(\mathbf{X})\right)
\end{aligned}
\tag{D.24}
$$

Note that β may vary for each output dimension, though in the original model it was assumed to be shared to aid computation.

Appendix E

LABGPLVM

The novel non-Gaussian likelihood BGPLVM model, the LABGPLVM proposed in Chapter 5, Section 5.6, builds upon the derivation in Section D. Various aspects of the derivation is used in the main text, however a more verbose derivation is again provided below for the interested reader.

E.1 Integration over latent function $p(\boldsymbol{s}|\mathbf{y})$

This section builds upon the derivation of the BGPLVM given in Section D, giving a more verbose derivation of the novel model proposed in Section 5.6.

The model treats the observations $\boldsymbol{S}$ as a corruption of $\mathbf{Y}$. This means that the bound given by Equation (D.24) can be used as a prior for the likelihood $p(\boldsymbol{S}|\mathbf{Y})$. This leaves one final integral to undertake, the integral of this prior, $\hat{p}(\mathbf{Y}|\mathbf{Z})$ with the likelihood, $p(\boldsymbol{S}|y)$. Since each output may have a different likelihood, we consider a single output dimension and perform j integrals,

$$p(\boldsymbol{s}_{:,j}|\mathbf{Z}) = \int p(\boldsymbol{s}_{:,j}|\mathbf{y}_{:,j})\hat{p}_j(\mathbf{y}_{:,j}|\mathbf{Z})\, d\mathbf{y}_{:,j} \tag{E.1}$$

if this likelihood $p(\boldsymbol{s}_{:,j}|\mathbf{y}_{:,j})$ is non-Gaussian, this integral will be intractable if and so an approximation is needed. In this work a Laplace approximation is proposed, though as discussed in Section 5.6 and Section 3.2 there are a there are a number of alternative approximation methods that could be used. Each output dimension j may also have its own β_j^{-1}, we will suppress this notation to maintain consistency with the BGPLVM bound.

We make an approximation for each dimension, and so where all $\mathbf{y}_{:,j}$ are observed nothing changes, otherwise only the observed values are retained, ${\psi_0}^{\mathcal{O}_j}$, ${\boldsymbol{\psi}_1}^{\mathcal{O}_j}$, ${\boldsymbol{\psi}_2}^{\mathcal{O}_j}$

etc, as in the main text of Section 5.5.1. This notation will be suppressed throughout for clarity but one should be aware that for each output $\boldsymbol{\psi}_1$ etc may vary.

$$
\begin{aligned}
&\log p(\boldsymbol{s}|\mathbf{Z}) \\
&\quad = \log \int p(\boldsymbol{s}|\mathbf{y}) p(\mathbf{y}|\mathbf{Z})\, d\mathbf{y} \\
&\quad \geq \log \int p(\boldsymbol{s}|\mathbf{y}) \exp(\log \hat{p}(\mathbf{y}|\mathbf{Z}))\, d\mathbf{y} \\
&\quad = \log \int p(\boldsymbol{s}|\mathbf{y}) \exp\Bigg(\sum_{j=1}^{p} \Big[\log \mathcal{N}\left(\mathbf{y}_{:,j}|\mathbf{0}, \hat{\mathbf{K}}\right) + \log |(\mathbf{K_{uu}} + \beta(\boldsymbol{\psi}_2 - {\boldsymbol{\psi}_1}^{\top}\boldsymbol{\psi}_1))^{-1}| \\
&\qquad + \log |\mathbf{K_{uu}}| - \frac{\beta}{2}\mathrm{tr}\left(\psi_0\right) + \frac{\beta}{2}\mathrm{tr}\left(\mathbf{K}_{\mathbf{uu}}^{-1}\boldsymbol{\psi}_2\right) \Big] - \mathrm{KL}\left(q(\mathbf{X}|\boldsymbol{\theta}_V) \,\|\, p(\mathbf{X})\right) \Bigg)\, d\mathbf{y} \\
&\quad = \sum_{j=1}^{p} \Big[\log |(\mathbf{K_{uu}} + \beta(\boldsymbol{\psi}_2 - {\boldsymbol{\psi}_1}^{\top}\boldsymbol{\psi}_1))^{-1}| + \log |\mathbf{K_{uu}}| - \frac{\beta}{2}\mathrm{tr}\left(\psi_0\right) + \frac{\beta}{2}\mathrm{tr}\left(\mathbf{K}_{\mathbf{uu}}^{-1}\boldsymbol{\psi}_2\right) \Big] \\
&\quad - \mathrm{KL}\left(q(\mathbf{X}|\boldsymbol{\theta}_V) \,\|\, p(\mathbf{X})\right) + \log \underbrace{\int p(\boldsymbol{S}|\mathbf{Y}) \prod_{j=1}^{p} \mathcal{N}\left(\mathbf{y}_{:,j}|\mathbf{0}, \hat{\mathbf{K}}\right)\, d\mathbf{y}}_{\text{Laplace approximation}}
\end{aligned}
$$

if the likelihood factorises over output dimensions we have

$$
\begin{aligned}
&\log p(\boldsymbol{s}|\mathbf{Z}) \\
&\quad = \sum_{j=1}^{p} \Big[\log |(\mathbf{K_{uu}} + \beta(\boldsymbol{\psi}_2 - {\boldsymbol{\psi}_1}^{\top}\boldsymbol{\psi}_1))^{-1}| + \log |\mathbf{K_{uu}}| - \frac{\beta}{2}\mathrm{tr}\left(\psi_0\right) + \frac{\beta}{2}\mathrm{tr}\left(\mathbf{K}_{\mathbf{uu}}^{-1}\boldsymbol{\psi}_2\right) \Big] \\
&\quad - \mathrm{KL}\left(q(\mathbf{X}|\boldsymbol{\theta}_V) \,\|\, p(\mathbf{X})\right) + \sum_{j=1}^{p} \log \underbrace{\int p(\boldsymbol{s}_{:,j}|\mathbf{y}_{:,j}) \mathcal{N}\left(\mathbf{y}_{:,j}|\mathbf{0}, \hat{\mathbf{K}}\right)\, d\mathbf{y}_{:,j}}_{\text{Laplace approximation}}
\end{aligned}
$$

Using the results of the Laplace approximation given in Section 3.2.1 the approximation can be made as follows,

$$
\begin{aligned}
&\log p(\boldsymbol{s}|\mathbf{Z}) \\
&\quad = \sum_{j=1}^{p} \Big[\log |(\mathbf{K_{uu}} + \beta(\boldsymbol{\psi}_2 - {\boldsymbol{\psi}_1}^{\top}\boldsymbol{\psi}_1))^{-1}| + \log |\mathbf{K_{uu}}| - \frac{\beta}{2}\mathrm{tr}\left(\psi_0\right) + \frac{\beta}{2}\mathrm{tr}\left(\mathbf{K}_{\mathbf{uu}}^{-1}\boldsymbol{\psi}_2\right) \Big] \\
&\quad - \mathrm{KL}\left(q(\mathbf{X}|\boldsymbol{\theta}_V) \,\|\, p(\mathbf{X})\right) + \sum_{j=1}^{p} \Big(- \log |\mathbf{I} + \hat{\mathbf{K}}\boldsymbol{W}_j| - \frac{1}{2}\hat{\mathbf{y}}_{:,j}^{\top}\hat{\mathbf{K}}^{-1}\hat{\mathbf{y}}_{:,j} + \log p(\boldsymbol{s}_{:,j}|\hat{\mathbf{y}}_{:,j}) \Big)
\end{aligned}
$$

where, $\hat{\mathbf{K}} = \beta^{-1}\mathbf{I}_j + \boldsymbol{\psi}_1(\mathbf{K_{uu}} + \beta(\boldsymbol{\psi}_2 - {\boldsymbol{\psi}_1}^{\top}\boldsymbol{\psi}_1))^{-1}{\boldsymbol{\psi}_1}^{\top}$.

For notational convenience, define $\boldsymbol{A} = \mathbf{K}_{\mathbf{uu}} + \beta(\boldsymbol{\psi}_2 - \boldsymbol{\psi}_1{}^\top \boldsymbol{\psi}_1)$,

$$\log p(\boldsymbol{s}|\mathbf{Z}) = \sum_{p=1}^{p} \Big[-\log|\boldsymbol{A}| - \log|\mathbf{K}_{\mathbf{uu}}^{-1}| - \frac{\beta}{2}\mathrm{tr}\left(\psi_0\right) + \frac{\beta}{2}\mathrm{tr}\left(\mathbf{K}_{\mathbf{uu}}^{-1}\psi_2\right) - \log|\mathbf{I} + \hat{\mathbf{K}}\boldsymbol{W}_j| \\ - \frac{1}{2}\hat{\mathbf{y}}_{:,j}^\top \hat{\mathbf{K}}^{-1}\hat{\mathbf{y}}_{:,j} + \log p(\boldsymbol{s}_{:,j}|\hat{\mathbf{y}}_{:,j}) \Big] - \mathrm{KL}\left(q(\mathbf{X}|\boldsymbol{\theta}_V) \,\|\, p(\mathbf{X})\right)$$

where, $\hat{\mathbf{K}} = \beta^{-1}\mathbf{I}_j + \boldsymbol{\psi}_1 \boldsymbol{A}^{-1} \boldsymbol{\psi}_1{}^\top$.

Note that since the Laplace approximation term is the only term that concerns $\boldsymbol{s}_{:,j}$, using the Laplace approximation also provides the posterior with respect to $\boldsymbol{s}_{:,j}$,

$$p(\mathbf{y}_{:,j}|\boldsymbol{s}_{:,j}) = \mathcal{N}\left(\mathbf{y}_{:,j}|\hat{\mathbf{y}}_{:,j}, (\hat{\mathbf{K}}^{-1} + \boldsymbol{W}_j)^{-1}\right) \tag{E.2}$$

This results will prove useful when we consider the posterior $p(\mathbf{u}_{:,j}|\boldsymbol{s}_{:,j})$, in Section E.3, with which we can make predictions. First we consider the models derivatives required for optimisation.

E.1.1 Laplace BGPLVM Derivatives

The model contains a number of parameters $\boldsymbol{\theta}_K, \boldsymbol{\theta}_L, \beta^{-1}$ that must be optimised. It is not possible to obtain a fixed point equation such that the gradient of the model is zero, and as such some method of gradient descent must be used. In order to implement with the derivatives with respect to $\boldsymbol{\theta}_K, \boldsymbol{\theta}_L, \beta^{-1}$ are required. The gradients for the kernel hyper-parameters, $\boldsymbol{\theta}_K$, can be found by using the chain rule $\frac{dL}{d\mathbf{K}_{\mathbf{uu}}}\frac{d\mathbf{K}_{\mathbf{uu}}}{d\boldsymbol{\theta}_K} + \frac{dL}{d\psi_0}\frac{d\psi_0}{d\boldsymbol{\theta}_K} + \frac{dL}{d\psi_1}\frac{d\psi_1}{d\boldsymbol{\theta}_K} + \frac{dL}{d\psi_2}\frac{d\psi_2}{d\boldsymbol{\theta}_K}$, the second term in each chain is specific to the kernel and we will assume for now that it is computable and focus on the first terms $\frac{dL}{d\mathbf{K}_{\mathbf{uu}}}, \frac{dL}{d\psi_0}, \frac{dL}{d\psi_1}, \frac{dL}{d\psi_2}, \frac{dL}{d\beta}$, where $L = \log p(\boldsymbol{s}|\mathbf{Z}, \boldsymbol{\theta}_L, \boldsymbol{\theta}_K)$. For calculating the derivatives for the variational parameters, $\frac{dL}{d\psi_0}, \frac{dL}{d\psi_1}, \frac{dL}{d\psi_2}$, the terms $\frac{d\psi_0}{d\boldsymbol{\theta}_V}, \frac{d\psi_1}{d\boldsymbol{\theta}_V}$, and $\frac{d\psi_2}{d\boldsymbol{\theta}_V}$ need to be computed for the respective kernel, examples for the ARD RBF kernel can be found in Gal and van der Wilk (2014).

In what follows are the key components of the partial derivatives required for applying the chain rule, in order to aid readers who wish to reimplement the model for their own uses.

To ease the derivations we will use the fact that we have a sum of may terms, and do each terms derivatives individually, as such we will number the terms

$$\begin{aligned}\log p(\boldsymbol{s}|\mathbf{Z}) = \sum_{j=1}^{p}\Bigg(& \underbrace{-\frac{1}{2}\hat{\mathbf{y}}_{:,j}^{\top}\hat{\mathbf{K}}^{-1}\hat{\mathbf{y}}_{:,j}}_{t_1} + \underbrace{\log p(\boldsymbol{s}_{:,j}|\hat{\mathbf{y}}_{:,j})}_{t_2} \underbrace{-\log|\mathbf{I}+\hat{\mathbf{K}}\boldsymbol{W}_j|}_{t_3} \underbrace{-\log|\boldsymbol{A}|}_{t_4} \\ & \underbrace{-\log|\mathbf{K}_{\mathbf{uu}}^{-1}|}_{t_5} \underbrace{-\frac{\beta}{2}\mathrm{Tr}(\psi_0-\mathbf{K}_{\mathbf{uu}}^{-1}\boldsymbol{\psi}_2)}_{t_6}\Bigg) \underbrace{-\mathrm{KL}\left(q(\mathbf{X}|\boldsymbol{\theta}_V)\,\|\,p(\mathbf{X})\right)}_{t_7}\end{aligned}$$

Most of the gradients are straightforward, however the Laplace approximation introduces a complication in that when we change $\boldsymbol{\theta}_K$, the mode $\hat{\mathbf{y}}$ also changes, these are known as *implicit* gradients, see Rasmussen and Williams (2006) to see how they are handled in the basic Laplace approximation with a Gaussian process prior. In general the gradients can be handled by splitting the gradients into two components, $\frac{dt}{d\boldsymbol{\theta}_K}+\sum_{i=1}^{n}\frac{dt}{d\hat{\mathbf{y}}_{ip}}\frac{d\hat{\mathbf{y}}_{ip}}{d\boldsymbol{\theta}_K}$, we refer to the former term as the *explicit* gradient, and the latter term the *implicit* gradient.

For convenience we will denote $\boldsymbol{\theta}_K=\{\mathbf{K}_{\mathbf{uu}},\psi_0,\boldsymbol{\psi}_1,\boldsymbol{\psi}_2,\beta\}$, this will help us derive derivatives that are simple to chain. Again note that in the case of missing data, these derivatives must be computed for each dimension.

The notation used throughout is in terms of infinitesimals. We abuse notation in the following way, by writing $\frac{d\boldsymbol{A}}{d\mathbf{K}_{\mathbf{uu}}}$ we are talking about how the function $\boldsymbol{A}$ changes as we make an infinitely small change in $\mathbf{K}_{\mathbf{uu}}$, whilst holding the other parameters of the function fixed. As such it describes a infinitesimal describing the change of the function in response to the change of another function.

$$\begin{aligned}\frac{d\boldsymbol{A}}{d\mathbf{K}_{\mathbf{uu}}} &= d\mathbf{K}_{\mathbf{uu}} \\ \frac{d\boldsymbol{A}}{d\psi_0} &= 0 \\ \frac{d\boldsymbol{A}}{d\boldsymbol{\psi}_1} &= -\frac{d\beta\boldsymbol{\psi}_1{}^{\top}\boldsymbol{\psi}_1}{\boldsymbol{\psi}_1} = -(\beta(d\boldsymbol{\psi}_1{}^{\top})\boldsymbol{\psi}_1+\beta\boldsymbol{\psi}_1{}^{\top}(d\boldsymbol{\psi}_1)) \\ \frac{d\boldsymbol{A}}{d\boldsymbol{\psi}_2} &= \beta(d\boldsymbol{\psi}_2) \\ \frac{d\boldsymbol{A}}{d\beta} &= (d\beta)(\boldsymbol{\psi}_2-\boldsymbol{\psi}_1{}^{\top}\boldsymbol{\psi}_1)\end{aligned}$$

and it follows that

$$
\begin{aligned}
\frac{d\hat{\mathbf{K}}}{d\mathbf{K}_{\mathbf{uu}}} &= \frac{d(\beta^{-1}\mathbf{I}_j + \boldsymbol{\psi}_1\boldsymbol{A}^{-1}\boldsymbol{\psi}_1{}^\top)}{d\mathbf{K}_{\mathbf{uu}}} = -\boldsymbol{\psi}_1\boldsymbol{A}^{-1}\frac{d\boldsymbol{A}}{d\mathbf{K}_{\mathbf{uu}}}\boldsymbol{A}^{-1}\boldsymbol{\psi}_1{}^\top = -\boldsymbol{\psi}_1\boldsymbol{A}^{-1}(d\mathbf{K}_{\mathbf{uu}})\boldsymbol{A}^{-1}\boldsymbol{\psi}_1{}^\top \\
\frac{d\hat{\mathbf{K}}}{d\boldsymbol{\psi}_1} &= \frac{d\boldsymbol{\psi}_1\boldsymbol{A}^{-1}}{d\boldsymbol{\psi}_1}\boldsymbol{\psi}_1{}^\top + \boldsymbol{\psi}_1\boldsymbol{A}^{-1}(d\boldsymbol{\psi}_1)^\top = (d\boldsymbol{\psi}_1)\boldsymbol{A}^{-1}\boldsymbol{\psi}_1{}^\top + \boldsymbol{\psi}_1\frac{d\boldsymbol{A}^{-1}}{d\boldsymbol{\psi}_1}\boldsymbol{\psi}_1{}^\top + \boldsymbol{\psi}_1\boldsymbol{A}^{-1}(d\boldsymbol{\psi}_1)^\top \\
&= (d\boldsymbol{\psi}_1)\boldsymbol{A}^{-1}\boldsymbol{\psi}_1{}^\top - \boldsymbol{\psi}_1\boldsymbol{A}^{-1}\frac{d\boldsymbol{A}}{d\boldsymbol{\psi}_1}\boldsymbol{A}^{-1}\boldsymbol{\psi}_1{}^\top + \boldsymbol{\psi}_1\boldsymbol{A}^{-1}(d\boldsymbol{\psi}_1)^\top \\
&= (d\boldsymbol{\psi}_1)\boldsymbol{A}^{-1}\boldsymbol{\psi}_1{}^\top + \boldsymbol{\psi}_1\boldsymbol{A}^{-1}\Big(\beta(d\boldsymbol{\psi}_1{}^\top)\boldsymbol{\psi}_1 + \beta\boldsymbol{\psi}_1{}^\top(d\boldsymbol{\psi}_1)\Big)\boldsymbol{A}^{-1}\boldsymbol{\psi}_1{}^\top + \boldsymbol{\psi}_1\boldsymbol{A}^{-1}(d\boldsymbol{\psi}_1)^\top \\
&= (d\boldsymbol{\psi}_1)\boldsymbol{A}^{-1}\boldsymbol{\psi}_1{}^\top + \boldsymbol{\psi}_1\boldsymbol{A}^{-1}\beta(d\boldsymbol{\psi}_1{}^\top)\boldsymbol{\psi}_1\boldsymbol{A}^{-1}\boldsymbol{\psi}_1{}^\top + \boldsymbol{\psi}_1\boldsymbol{A}^{-1}\beta\boldsymbol{\psi}_1{}^\top(d\boldsymbol{\psi}_1)\boldsymbol{A}^{-1}\boldsymbol{\psi}_1{}^\top \\
&\quad + \boldsymbol{\psi}_1\boldsymbol{A}^{-1}(d\boldsymbol{\psi}_1)^\top \\
&= (d\boldsymbol{\psi}_1)\boldsymbol{A}^{-1}\boldsymbol{\psi}_1{}^\top + \Big(\boldsymbol{\psi}_1\boldsymbol{A}^{-1}\boldsymbol{\psi}_1{}^\top(d\boldsymbol{\psi}_1)\beta\boldsymbol{A}^{-1}\boldsymbol{\psi}_1{}^\top\Big)^\top + \boldsymbol{\psi}_1\boldsymbol{A}^{-1}\beta\boldsymbol{\psi}_1{}^\top(d\boldsymbol{\psi}_1)\boldsymbol{A}^{-1}\boldsymbol{\psi}_1{}^\top \\
&\quad + \Big((d\boldsymbol{\psi}_1)\boldsymbol{A}^{-1}\boldsymbol{\psi}_1{}^\top\Big)^\top \\
&= \underbrace{(d\boldsymbol{\psi}_1)\boldsymbol{A}^{-1}\boldsymbol{\psi}_1{}^\top}_{G} + \underbrace{\Big(\boldsymbol{\psi}_1\boldsymbol{A}^{-1}\boldsymbol{\psi}_1{}^\top(d\boldsymbol{\psi}_1)\beta\boldsymbol{A}^{-1}\boldsymbol{\psi}_1{}^\top\Big)^\top}_{E^\top} \\
&\quad + \underbrace{\boldsymbol{\psi}_1\boldsymbol{A}^{-1}\beta\boldsymbol{\psi}_1{}^\top(d\boldsymbol{\psi}_1)\boldsymbol{A}^{-1}\boldsymbol{\psi}_1{}^\top}_{E} + \underbrace{\Big((d\boldsymbol{\psi}_1)\boldsymbol{A}^{-1}\boldsymbol{\psi}_1{}^\top\Big)^\top}_{G^\top} \\
&= (G + G^\top) + (E + E^\top) \\
\frac{d\hat{\mathbf{K}}}{d\boldsymbol{\psi}_2} &= -\boldsymbol{\psi}_1\boldsymbol{A}^{-1}\frac{d\boldsymbol{A}}{d\boldsymbol{\psi}_2}\boldsymbol{A}^{-1}\boldsymbol{\psi}_1{}^\top = -\boldsymbol{\psi}_1\boldsymbol{A}^{-1}\beta(d\boldsymbol{\psi}_2)\boldsymbol{A}^{-1}\boldsymbol{\psi}_1{}^\top \\
\frac{d\hat{\mathbf{K}}}{d\psi_0} &= 0 \\
\frac{d\hat{\mathbf{K}}}{d\beta} &= -\beta^{-1}(d\beta)\beta^{-1} + \boldsymbol{\psi}_1\frac{d\boldsymbol{A}^{-1}}{d\beta}\boldsymbol{\psi}_1{}^\top \\
&= -\beta^{-1}(d\beta)\beta^{-1} - \boldsymbol{\psi}_1\boldsymbol{A}^{-1}(d\beta)(\boldsymbol{\psi}_2 - \boldsymbol{\psi}_1{}^\top\boldsymbol{\psi}_1)\boldsymbol{A}^{-1}\boldsymbol{\psi}_1{}^\top \\
&= -(\beta^{-2} + \boldsymbol{\psi}_1\boldsymbol{A}^{-1}(\boldsymbol{\psi}_2 - \boldsymbol{\psi}_1{}^\top\boldsymbol{\psi}_1)\boldsymbol{A}^{-1}\boldsymbol{\psi}_1{}^\top)(d\beta)
\end{aligned}
$$

It is convenient to obtain the derivatives by manipulating the infinitesimals the following form of $\frac{dt}{d\boldsymbol{\theta}} = \operatorname{tr}\left(X(d\boldsymbol{\theta})\right)$. For further details see on this method of obtaining derivatives, see Minka (2000). The derivatives for t_6, t_7 are equivalent to the BGPLVM, see Damianou (2015).

t_1 derivatives

$$\frac{dt_1}{d\boldsymbol{\theta}_K} = \frac{d(-\frac{1}{2}\hat{\mathbf{y}}^\top\hat{\mathbf{K}}^{-1}\hat{\mathbf{y}})}{d\boldsymbol{\theta}_K} = \frac{1}{2}\hat{\mathbf{y}}^\top\hat{\mathbf{K}}^{-1}(d\hat{\mathbf{K}})\hat{\mathbf{K}}^{-1}\hat{\mathbf{y}} = \frac{1}{2}\mathrm{Tr}\Big(\hat{\mathbf{K}}^{-1}\hat{\mathbf{y}}\hat{\mathbf{y}}^\top\hat{\mathbf{K}}^{-1}(d\hat{\mathbf{K}})\Big)$$

$$\begin{aligned}
\frac{dt_1}{d\boldsymbol{\psi}_1} &= \frac{1}{2}\mathrm{Tr}\Big(\hat{\mathbf{K}}^{-1}\hat{\mathbf{y}}\hat{\mathbf{y}}^\top\hat{\mathbf{K}}^{-1}(d\hat{\mathbf{K}})\Big)\\
&= \frac{1}{2}\mathrm{Tr}\Big(\hat{\mathbf{K}}^{-1}\hat{\mathbf{y}}\hat{\mathbf{y}}^\top\hat{\mathbf{K}}^{-1}(G + G^\top + E + E^\top)\Big)\\
&= \frac{1}{2}\mathrm{Tr}\Big(\hat{\mathbf{K}}^{-1}\hat{\mathbf{y}}\hat{\mathbf{y}}^\top\hat{\mathbf{K}}^{-1}G\Big) + \frac{1}{2}\mathrm{Tr}\Big(\hat{\mathbf{K}}^{-1}\hat{\mathbf{y}}\hat{\mathbf{y}}^\top\hat{\mathbf{K}}^{-1}G^\top\Big) + \frac{1}{2}\mathrm{Tr}\Big(\hat{\mathbf{K}}^{-1}\hat{\mathbf{y}}\hat{\mathbf{y}}^\top\hat{\mathbf{K}}^{-1}E\Big)\\
&\quad + \frac{1}{2}\mathrm{Tr}\Big(\hat{\mathbf{K}}^{-1}\hat{\mathbf{y}}\hat{\mathbf{y}}^\top\hat{\mathbf{K}}^{-1}E^\top\Big)\\
&= \frac{1}{2}\mathrm{Tr}\Big(\hat{\mathbf{K}}^{-1}\hat{\mathbf{y}}\hat{\mathbf{y}}^\top\hat{\mathbf{K}}^{-1}G\Big) + \frac{1}{2}\mathrm{Tr}\Big(\hat{\mathbf{K}}^{-1}\hat{\mathbf{y}}\hat{\mathbf{y}}^\top\hat{\mathbf{K}}^{-1}G\Big) + \frac{1}{2}\mathrm{Tr}\Big(\hat{\mathbf{K}}^{-1}\hat{\mathbf{y}}\hat{\mathbf{y}}^\top\hat{\mathbf{K}}^{-1}E\Big)\\
&\quad + \frac{1}{2}\mathrm{Tr}\Big(\hat{\mathbf{K}}^{-1}\hat{\mathbf{y}}\hat{\mathbf{y}}^\top\hat{\mathbf{K}}^{-1}E\Big)\\
&= \mathrm{Tr}\Big(\hat{\mathbf{K}}^{-1}\hat{\mathbf{y}}\hat{\mathbf{y}}^\top\hat{\mathbf{K}}^{-1}G\Big) + \mathrm{Tr}\Big(\hat{\mathbf{K}}^{-1}\hat{\mathbf{y}}\hat{\mathbf{y}}^\top\hat{\mathbf{K}}^{-1}E\Big)\\
&= \mathrm{Tr}\Big(\hat{\mathbf{K}}^{-1}\hat{\mathbf{y}}\hat{\mathbf{y}}^\top\hat{\mathbf{K}}^{-1}(d\boldsymbol{\psi}_1)\boldsymbol{A}^{-1}{\boldsymbol{\psi}_1}^\top\Big) + \mathrm{Tr}\Big(\hat{\mathbf{K}}^{-1}\hat{\mathbf{y}}\hat{\mathbf{y}}^\top\hat{\mathbf{K}}^{-1}\boldsymbol{\psi}_1\boldsymbol{A}^{-1}\beta{\boldsymbol{\psi}_1}^\top(d\boldsymbol{\psi}_1)\boldsymbol{A}^{-1}{\boldsymbol{\psi}_1}^\top\Big)\\
&= \mathrm{Tr}\Big(\boldsymbol{A}^{-1}{\boldsymbol{\psi}_1}^\top\hat{\mathbf{K}}^{-1}\hat{\mathbf{y}}\hat{\mathbf{y}}^\top\hat{\mathbf{K}}^{-1}(d\boldsymbol{\psi}_1)\Big) + \mathrm{Tr}\Big(\boldsymbol{A}^{-1}{\boldsymbol{\psi}_1}^\top\hat{\mathbf{K}}^{-1}\hat{\mathbf{y}}\hat{\mathbf{y}}^\top\hat{\mathbf{K}}^{-1}\boldsymbol{\psi}_1\boldsymbol{A}^{-1}\beta{\boldsymbol{\psi}_1}^\top(d\boldsymbol{\psi}_1)\Big)\\
&= \mathrm{Tr}\Big(\boldsymbol{A}^{-1}{\boldsymbol{\psi}_1}^\top\hat{\mathbf{K}}^{-1}\hat{\mathbf{y}}\hat{\mathbf{y}}^\top\hat{\mathbf{K}}^{-1}(\mathbf{I} + \boldsymbol{\psi}_1\boldsymbol{A}^{-1}\beta{\boldsymbol{\psi}_1}^\top)(d\boldsymbol{\psi}_1)\Big)
\end{aligned}$$

$$\begin{aligned}
\frac{dt_1}{d\boldsymbol{\psi}_2} &= \frac{1}{2}\mathrm{Tr}\Big(\hat{\mathbf{K}}^{-1}\hat{\mathbf{y}}\hat{\mathbf{y}}^\top\hat{\mathbf{K}}^{-1}(d\hat{\mathbf{K}})\Big)\\
&= \frac{1}{2}\mathrm{Tr}\Big(\hat{\mathbf{K}}^{-1}\hat{\mathbf{y}}\hat{\mathbf{y}}^\top\hat{\mathbf{K}}^{-1}(-\boldsymbol{\psi}_1\boldsymbol{A}^{-1}\beta(d\boldsymbol{\psi}_2)\boldsymbol{A}^{-1}{\boldsymbol{\psi}_1}^\top)\Big)\\
&= -\frac{1}{2}\mathrm{Tr}\Big(\boldsymbol{A}^{-1}{\boldsymbol{\psi}_1}^\top\hat{\mathbf{K}}^{-1}\hat{\mathbf{y}}\hat{\mathbf{y}}^\top\hat{\mathbf{K}}^{-1}\boldsymbol{\psi}_1\boldsymbol{A}^{-1}\beta(d\boldsymbol{\psi}_2)\Big)
\end{aligned}$$

$$\begin{aligned}
\frac{dt_1}{d\beta} &= \frac{1}{2}\mathrm{Tr}\Big(\hat{\mathbf{K}}^{-1}\hat{\mathbf{y}}\hat{\mathbf{y}}^\top\hat{\mathbf{K}}^{-1}(d\hat{\mathbf{K}})\Big)\\
&= -\frac{1}{2}\mathrm{Tr}\Big(\hat{\mathbf{K}}^{-1}\hat{\mathbf{y}}\hat{\mathbf{y}}^\top\hat{\mathbf{K}}^{-1}(\beta^{-2} + \boldsymbol{\psi}_1\boldsymbol{A}^{-1}(\boldsymbol{\psi}_2 - {\boldsymbol{\psi}_1}^\top\boldsymbol{\psi}_1)\boldsymbol{A}^{-1}{\boldsymbol{\psi}_1}^\top)(d\beta)\Big)
\end{aligned}$$

t_2 derivatives

$$\frac{dt_2}{d\boldsymbol{\theta}_K} = \frac{d\log p(\boldsymbol{s}_{:,j}|\hat{\mathbf{y}}_{:,j})}{d\boldsymbol{\theta}_K} = 0$$

t_3 derivatives

$$\begin{aligned}\frac{dt_3}{d\boldsymbol{\theta}_K}_{\text{explicit}} &= \frac{d(-\log|\mathbf{I}+\hat{\mathbf{K}}\boldsymbol{W}|)}{d\boldsymbol{\theta}_K} \\ &= -\mathrm{Tr}\Big((\mathbf{I}+\hat{\mathbf{K}}\boldsymbol{W})^{-1}(d\hat{\mathbf{K}})\boldsymbol{W}\Big) \\ &= -\mathrm{Tr}\Big((\boldsymbol{W}^{-1}+\hat{\mathbf{K}})^{-1}(d\hat{\mathbf{K}})\Big)\end{aligned}$$

Implicit Derivatives

The implicit derivatives arise from the fact that as the kernel parameters change, the mode also changes, this can be seen as a chain of the $\frac{dt}{d\hat{\mathbf{y}}}$ and $(d\hat{\mathbf{y}})$. Things are complicated further by the fact that $(d\hat{\mathbf{y}})$ could be invalid as $\hat{\mathbf{y}}$ is a vector and $d\boldsymbol{\theta}_K$ could be a matrix. Derivatives of vectors by matrices is mathematically invalid. Instead we must obtain the derivative for each element of the vector separately. As such we write the implicit derivatives as

$$\frac{dt}{d\boldsymbol{\theta}_K} = \sum_{i=1}^{n}\frac{dt}{d\hat{\mathbf{y}}_i}(d\hat{\mathbf{y}}_i)$$

The term $\frac{dt_3}{d\hat{\mathbf{y}}_i}$ is straightforward to compute, since $\frac{dt_3}{d\hat{\mathbf{y}}}$ is not illegal to compute in full, and since $\frac{dt_3}{d\hat{\mathbf{y}}} = [\frac{dt_3}{d\hat{\mathbf{y}}_1}, \frac{dt_3}{d\hat{\mathbf{y}}_1}, \cdots, \frac{dt_3}{d\hat{\mathbf{y}}_n}]^\top$

$$
\begin{aligned}
\frac{dt_3}{d\hat{\mathbf{y}}_i} &= -\frac{d\log|\mathbf{I}+\hat{\mathbf{K}}\boldsymbol{W}|}{d\hat{\mathbf{y}}_i} \\
&= -\mathrm{Tr}\Big((\mathbf{I}+\hat{\mathbf{K}}\boldsymbol{W})^{-1}\hat{\mathbf{K}}\frac{d\boldsymbol{W}}{d\hat{\mathbf{y}}_i}\Big) \\
&= -\mathrm{Tr}\Big((\hat{\mathbf{K}}^{-1}+\boldsymbol{W})^{-1}\frac{d\boldsymbol{W}}{d\hat{\mathbf{y}}_i}\Big) \\
&= -\mathrm{Tr}\Big((\hat{\mathbf{K}}^{-1}+\boldsymbol{W})^{-1}(-\frac{d^3\log p(\boldsymbol{s}|\hat{\mathbf{y}})}{d\hat{\mathbf{y}}_i^3})\Big)
\end{aligned}
$$

The latter term, $(d\hat{\mathbf{y}})$ is more difficult since $\boldsymbol{\theta}_K$ could be a matrix, making the operation invalid mathematically.

We can write $(d\hat{\mathbf{y}})$ using a vector $\mathbf{e}_i \triangleq [0,0,\ldots 1,0,0]^\top$ where the position of the 1 is at i. As such we can write the infinitesimal

$$
\begin{aligned}
d\hat{\mathbf{y}}_i &= d(\mathbf{e}_i^\top d\hat{\mathbf{y}}) \\
&= \mathbf{e}_i^\top (d\hat{\mathbf{y}})
\end{aligned}
$$

Since $\hat{\mathbf{y}} = \hat{\mathbf{K}}\frac{d\log p(\boldsymbol{s}_{:,j}|\hat{\mathbf{y}}_{:,j})}{d\hat{\mathbf{y}}_{:,j}}$, for $\boldsymbol{\psi}_2$, following a similar derivation as Rasmussen and Williams (2006),

$$
\begin{aligned}
d\hat{\mathbf{y}}_i &= \mathbf{e}_i^\top (d\hat{\mathbf{K}})\frac{d\log p(\boldsymbol{s}_{:,j}|\hat{\mathbf{y}}_{:,j})}{d\hat{\mathbf{y}}_{:,j}} + \mathbf{e}_i^\top\hat{\mathbf{K}}\Big(d\frac{d\log p(\boldsymbol{s}_{:,j}|\hat{\mathbf{y}}_{:,j})}{d\hat{\mathbf{y}}_{:,j}}\Big) \\
d\hat{\mathbf{y}}_i &= \mathbf{e}_i^\top (\mathbf{I}+\hat{\mathbf{K}}\boldsymbol{W})^{-1}(d\hat{\mathbf{K}})\frac{d\log p(\boldsymbol{s}_{:,j}|\hat{\mathbf{y}}_{:,j})}{d\hat{\mathbf{y}}_{:,j}}
\end{aligned}
$$

For $\boldsymbol{\psi}_2$

$$
\begin{aligned}
\frac{d\hat{\mathbf{y}}_i}{d\boldsymbol{\psi}_2} &= \mathbf{e}_i^\top (\mathbf{I}+\hat{\mathbf{K}}\boldsymbol{W})^{-1}(d\hat{\mathbf{K}})\frac{d\log p(\boldsymbol{s}_{:,j}|\hat{\mathbf{y}}_{:,j})}{d\hat{\mathbf{y}}_{:,j}} \\
&= -\mathbf{e}_i^\top (\mathbf{I}+\hat{\mathbf{K}}\boldsymbol{W})^{-1}\boldsymbol{\psi}_1\boldsymbol{A}^{-1}\beta(d\boldsymbol{\psi}_2)\boldsymbol{A}^{-1}\boldsymbol{\psi}_1{}^\top\frac{d\log p(\boldsymbol{s}_{:,j}|\hat{\mathbf{y}}_{:,j})}{d\hat{\mathbf{y}}_{:,j}} \\
&= \mathrm{Tr}\Big(-\boldsymbol{A}^{-1}\boldsymbol{\psi}_1{}^\top\frac{d\log p(\boldsymbol{s}_{:,j}|\hat{\mathbf{y}}_{:,j})}{d\hat{\mathbf{y}}_{:,j}}\mathbf{e}_i^\top(\mathbf{I}+\hat{\mathbf{K}}\boldsymbol{W})^{-1}\boldsymbol{\psi}_1\boldsymbol{A}^{-1}\beta(d\boldsymbol{\psi}_2)\Big)
\end{aligned}
$$

Then use the chain rule

$$\begin{aligned}
\frac{dt_3}{d\boldsymbol{\psi}_2} &= \sum_{i=1}^{n} \frac{dt_3}{d\hat{\mathbf{y}}_i}\frac{d\hat{\mathbf{y}}_i}{d\boldsymbol{\psi}_2} \\
&= \sum_{i=1}^{n} \mathrm{Tr}\Big(\frac{dt_3}{d\hat{\mathbf{y}}_i}\Big)\mathrm{Tr}\Big(- \boldsymbol{A}^{-1}\boldsymbol{\psi}_1{}^{\top}\frac{d\log p(\boldsymbol{s}_{:,j}|\hat{\mathbf{y}}_{:,j})}{d\hat{\mathbf{y}}_{:,j}}\mathbf{e}_i^{\top}(\mathbf{I}+\hat{\mathbf{K}}\boldsymbol{W})^{-1}\boldsymbol{\psi}_1\boldsymbol{A}^{-1}\beta(d\boldsymbol{\psi}_2)\Big) \\
&= -\sum_{i=1}^{n} \mathrm{Tr}\Big(\frac{dt_3}{d\hat{\mathbf{y}}_i}\boldsymbol{A}^{-1}\boldsymbol{\psi}_1{}^{\top}\frac{d\log p(\boldsymbol{s}_{:,j}|\hat{\mathbf{y}}_{:,j})}{d\hat{\mathbf{y}}_{:,j}}\mathbf{e}_i^{\top}(\mathbf{I}+\hat{\mathbf{K}}\boldsymbol{W})^{-1}\boldsymbol{\psi}_1\boldsymbol{A}^{-1}\beta(d\boldsymbol{\psi}_2)\Big) \\
&= -\sum_{i=1}^{n} \mathrm{Tr}\Big(\boldsymbol{A}^{-1}\boldsymbol{\psi}_1{}^{\top}\frac{d\log p(\boldsymbol{s}_{:,j}|\hat{\mathbf{y}}_{:,j})}{d\hat{\mathbf{y}}_{:,j}}\Big[\mathbf{e}_i^{\top}\frac{dt_3}{d\hat{\mathbf{y}}_i}\Big](\mathbf{I}+\hat{\mathbf{K}}\boldsymbol{W})^{-1}\boldsymbol{\psi}_1\boldsymbol{A}^{-1}\beta(d\boldsymbol{\psi}_2)\Big) \\
&\qquad\qquad\text{(Since it is a scalar value)} \\
&= -\mathrm{Tr}\Big(\boldsymbol{A}^{-1}\boldsymbol{\psi}_1{}^{\top}\frac{d\log p(\boldsymbol{s}_{:,j}|\hat{\mathbf{y}}_{:,j})}{d\hat{\mathbf{y}}_{:,j}}\Big[\sum_{i=1}^{n}\mathbf{e}_i^{\top}\frac{dt_3}{d\hat{\mathbf{y}}_i}\Big](\mathbf{I}+\hat{\mathbf{K}}\boldsymbol{W})^{-1}\boldsymbol{\psi}_1\boldsymbol{A}^{-1}\beta(d\boldsymbol{\psi}_2)\Big) \\
&\qquad\qquad\text{(Other terms independent of } i\text{)} \\
&= -\mathrm{Tr}\Big(\boldsymbol{A}^{-1}\boldsymbol{\psi}_1{}^{\top}\frac{d\log p(\boldsymbol{s}_{:,j}|\hat{\mathbf{y}}_{:,j})}{d\hat{\mathbf{y}}_{:,j}}\Big[\frac{dt_3}{d\hat{\mathbf{y}}}\Big]^{\top}(\mathbf{I}+\hat{\mathbf{K}}\boldsymbol{W})^{-1}\boldsymbol{\psi}_1\boldsymbol{A}^{-1}\beta(d\boldsymbol{\psi}_2)\Big)
\end{aligned}$$

The final line arises from $\mathbf{e}_i^{\top}$ picking out each element through the loop and adding it, providing $\frac{dt_3}{d\hat{\mathbf{y}}}$ in full.

Now repeat this same method for the other terms:

For $\boldsymbol{\psi}_1$

$$
\begin{aligned}
\frac{d\hat{\mathbf{y}}_i}{d\boldsymbol{\psi}_1} &= \mathbf{e}_i^\top(\mathbf{I}+\hat{\mathbf{K}}\boldsymbol{W})^{-1}(d\hat{\mathbf{K}})\frac{d\log p(\boldsymbol{s}_{:,j}|\hat{\mathbf{y}}_{:,j})}{d\hat{\mathbf{y}}_{:,j}}\\
\frac{d\hat{\mathbf{y}}_i}{d\boldsymbol{\psi}_1} &= \mathbf{e}_i^\top(\mathbf{I}+\hat{\mathbf{K}}\boldsymbol{W})^{-1}(G+G^\top+E+E^\top)\frac{d\log p(\boldsymbol{s}_{:,j}|\hat{\mathbf{y}}_{:,j})}{d\hat{\mathbf{y}}_{:,j}}\\
&= \mathrm{Tr}\Big(\mathbf{e}_i^\top(\mathbf{I}+\hat{\mathbf{K}}\boldsymbol{W})^{-1}\Big[(d\boldsymbol{\psi}_1)\boldsymbol{A}^{-1}{\boldsymbol{\psi}_1}^\top\Big]\frac{d\log p(\boldsymbol{s}_{:,j}|\hat{\mathbf{y}}_{:,j})}{d\hat{\mathbf{y}}_{:,j}}\Big)\\
&+ \mathrm{Tr}\Big(\mathbf{e}_i^\top(\mathbf{I}+\hat{\mathbf{K}}\boldsymbol{W})^{-1}\Big[(d\boldsymbol{\psi}_1)\boldsymbol{A}^{-1}{\boldsymbol{\psi}_1}^\top\Big]^\top\frac{d\log p(\boldsymbol{s}_{:,j}|\hat{\mathbf{y}}_{:,j})}{d\hat{\mathbf{y}}_{:,j}}\Big)\\
&+ \mathrm{Tr}\Big(\mathbf{e}_i^\top(\mathbf{I}+\hat{\mathbf{K}}\boldsymbol{W})^{-1}\Big[\boldsymbol{\psi}_1\boldsymbol{A}^{-1}\beta{\boldsymbol{\psi}_1}^\top(d\boldsymbol{\psi}_1)\boldsymbol{A}^{-1}{\boldsymbol{\psi}_1}^\top\Big]\frac{d\log p(\boldsymbol{s}_{:,j}|\hat{\mathbf{y}}_{:,j})}{d\hat{\mathbf{y}}_{:,j}}\Big)\\
&+ \mathrm{Tr}\Big(\mathbf{e}_i^\top(\mathbf{I}+\hat{\mathbf{K}}\boldsymbol{W})^{-1}\Big[\boldsymbol{\psi}_1\boldsymbol{A}^{-1}{\boldsymbol{\psi}_1}^\top(d\boldsymbol{\psi}_1)\beta\boldsymbol{A}^{-1}{\boldsymbol{\psi}_1}^\top\Big]^\top\frac{d\log p(\boldsymbol{s}_{:,j}|\hat{\mathbf{y}}_{:,j})}{d\hat{\mathbf{y}}_{:,j}}\Big)\\
&\qquad\qquad\qquad\qquad\text{(Now transpose everything in terms with } {\boldsymbol{\psi}_1}^\top\text{)}\\
&= \mathrm{Tr}\Big(\boldsymbol{A}^{-1}{\boldsymbol{\psi}_1}^\top\frac{d\log p(\boldsymbol{s}_{:,j}|\hat{\mathbf{y}}_{:,j})}{d\hat{\mathbf{y}}_{:,j}}\mathbf{e}_i^\top(\mathbf{I}+\hat{\mathbf{K}}\boldsymbol{W})^{-1}(d\boldsymbol{\psi}_1)\Big)\\
&+ \mathrm{Tr}\Big(\frac{d\log p(\boldsymbol{s}_{:,j}|\hat{\mathbf{y}}_{:,j})}{d\hat{\mathbf{y}}_{:,j}}^\top\Big[(d\boldsymbol{\psi}_1)\boldsymbol{A}^{-1}{\boldsymbol{\psi}_1}^\top\Big](\mathbf{I}+\hat{\mathbf{K}}\boldsymbol{W})^{-1\top}\mathbf{e}_i\Big)\\
&+ \mathrm{Tr}\Big(\boldsymbol{A}^{-1}{\boldsymbol{\psi}_1}^\top\frac{d\log p(\boldsymbol{s}_{:,j}|\hat{\mathbf{y}}_{:,j})}{d\hat{\mathbf{y}}_{:,j}}\mathbf{e}_i^\top(\mathbf{I}+\hat{\mathbf{K}}\boldsymbol{W})^{-1}\boldsymbol{\psi}_1\boldsymbol{A}^{-1}\beta{\boldsymbol{\psi}_1}^\top(d\boldsymbol{\psi}_1)\Big)\\
&+ \mathrm{Tr}\Big(\frac{d\log p(\boldsymbol{s}_{:,j}|\hat{\mathbf{y}}_{:,j})}{d\hat{\mathbf{y}}_{:,j}}^\top\Big[\boldsymbol{\psi}_1\boldsymbol{A}^{-1}{\boldsymbol{\psi}_1}^\top(d\boldsymbol{\psi}_1)\beta\boldsymbol{A}^{-1}{\boldsymbol{\psi}_1}^\top\Big](\mathbf{I}+\hat{\mathbf{K}}\boldsymbol{W})^{-1\top}\mathbf{e}_i\Big)\\
&\qquad\qquad\qquad\qquad\text{(Now rotate for } d\boldsymbol{\psi}_1\text{)}\\
&= \mathrm{Tr}\Big(\boldsymbol{A}^{-1}{\boldsymbol{\psi}_1}^\top\frac{d\log p(\boldsymbol{s}_{:,j}|\hat{\mathbf{y}}_{:,j})}{d\hat{\mathbf{y}}_{:,j}}\mathbf{e}_i^\top(\mathbf{I}+\hat{\mathbf{K}}\boldsymbol{W})^{-1}(d\boldsymbol{\psi}_1)\Big) && (\hat{A})\\
&+ \mathrm{Tr}\Big(\boldsymbol{A}^{-1}{\boldsymbol{\psi}_1}^\top(\mathbf{I}+\hat{\mathbf{K}}\boldsymbol{W})^{-1\top}\mathbf{e}_i\frac{d\log p(\boldsymbol{s}_{:,j}|\hat{\mathbf{y}}_{:,j})}{d\hat{\mathbf{y}}_{:,j}}^\top(d\boldsymbol{\psi}_1)\Big) && (\hat{B}^\top)\\
&+ \mathrm{Tr}\Big(\boldsymbol{A}^{-1}{\boldsymbol{\psi}_1}^\top\frac{d\log p(\boldsymbol{s}_{:,j}|\hat{\mathbf{y}}_{:,j})}{d\hat{\mathbf{y}}_{:,j}}\mathbf{e}_i^\top(\mathbf{I}+\hat{\mathbf{K}}\boldsymbol{W})^{-1}\boldsymbol{\psi}_1\boldsymbol{A}^{-1}\beta{\boldsymbol{\psi}_1}^\top(d\boldsymbol{\psi}_1)\Big) && (\hat{C})\\
&+ \mathrm{Tr}\Big(\beta\boldsymbol{A}^{-1}{\boldsymbol{\psi}_1}^\top(\mathbf{I}+\hat{\mathbf{K}}\boldsymbol{W})^{-1\top}\mathbf{e}_i\frac{d\log p(\boldsymbol{s}_{:,j}|\hat{\mathbf{y}}_{:,j})}{d\hat{\mathbf{y}}_{:,j}}^\top\boldsymbol{\psi}_1\boldsymbol{A}^{-1}{\boldsymbol{\psi}_1}^\top(d\boldsymbol{\psi}_1)\Big) && (\hat{D}^\top)
\end{aligned}
$$

Apply the chain rule,

$$\begin{aligned}
\frac{dt_3}{d\boldsymbol{\psi}_1} &= \sum_{i=1}^{n} \frac{dt_3}{d\hat{\mathbf{y}}_i} \frac{d\hat{\mathbf{y}}_i}{d\boldsymbol{\psi}_1} \\
&= \sum_{i=1}^{n} \mathrm{Tr}\Big(\frac{dt_3}{d\hat{\mathbf{y}}_i} A\Big) + \sum_{i=1}^{n} \mathrm{Tr}\Big(\frac{dt_3}{d\hat{\mathbf{y}}_i} B\Big) + \sum_{i=1}^{n} \mathrm{Tr}\Big(\frac{dt_3}{d\hat{\mathbf{y}}_i} C\Big) + \sum_{i=1}^{n} \mathrm{Tr}\Big(\frac{dt_3}{d\hat{\mathbf{y}}_i} D\Big) \\
&= \mathrm{Tr}\Big(\boldsymbol{A}^{-1}{\boldsymbol{\psi}_1}^{\top} \frac{d\log p(\boldsymbol{s}_{:,j}|\hat{\mathbf{y}}_{:,j})}{d\hat{\mathbf{y}}_{:,j}} \Big[\sum_{i=1}^{n} \mathbf{e}_i^{\top} \frac{dt_3}{d\hat{\mathbf{y}}_i}\Big] (\mathbf{I} + \hat{\mathbf{K}}\boldsymbol{W})^{-1} (d\boldsymbol{\psi}_1)\Big) \\
&+ \mathrm{Tr}\Big(\boldsymbol{A}^{-1}{\boldsymbol{\psi}_1}^{\top} (\mathbf{I} + \hat{\mathbf{K}}\boldsymbol{W})^{-1\top} \Big[\sum_{i=1}^{n} \frac{dt_3}{d\hat{\mathbf{y}}_i}^{\top} \mathbf{e}_i\Big] \frac{d\log p(\boldsymbol{s}_{:,j}|\hat{\mathbf{y}}_{:,j})}{d\hat{\mathbf{y}}_{:,j}}^{\top} (d\boldsymbol{\psi}_1)\Big) \\
&+ \mathrm{Tr}\Big(\boldsymbol{A}^{-1}{\boldsymbol{\psi}_1}^{\top} \frac{d\log p(\boldsymbol{s}_{:,j}|\hat{\mathbf{y}}_{:,j})}{d\hat{\mathbf{y}}_{:,j}} \Big[\sum_{i=1}^{n} \mathbf{e}_i^{\top} \frac{dt_3}{d\hat{\mathbf{y}}_i}\Big] (\mathbf{I} + \hat{\mathbf{K}}\boldsymbol{W})^{-1} \boldsymbol{\psi}_1 \boldsymbol{A}^{-1} \beta {\boldsymbol{\psi}_1}^{\top} (d\boldsymbol{\psi}_1)\Big) \\
&+ \mathrm{Tr}\Big(\beta \boldsymbol{A}^{-1}{\boldsymbol{\psi}_1}^{\top} (\mathbf{I} + \hat{\mathbf{K}}\boldsymbol{W})^{-1\top} \Big[\sum_{i=1}^{n} \frac{dt_3}{d\hat{\mathbf{y}}_i}^{\top} \mathbf{e}_i\Big] \frac{d\log p(\boldsymbol{s}_{:,j}|\hat{\mathbf{y}}_{:,j})}{d\hat{\mathbf{y}}_{:,j}}^{\top} \boldsymbol{\psi}_1 \boldsymbol{A}^{-1} {\boldsymbol{\psi}_1}^{\top} (d\boldsymbol{\psi}_1)\Big) \\
&\qquad\qquad \Big(\text{Note the use of trace of the scalar } \tfrac{t_3}{d\hat{\mathbf{y}}_i} \text{ and } \textstyle\sum_{i=1}^{n} \mathbf{e}_i^{\top} \tfrac{dt}{d\hat{\mathbf{y}}_i} = \tfrac{dt}{d\hat{\mathbf{y}}}^{\top}\Big) \\
&= \mathrm{Tr}\Big(\boldsymbol{A}^{-1}{\boldsymbol{\psi}_1}^{\top} \frac{d\log p(\boldsymbol{s}_{:,j}|\hat{\mathbf{y}}_{:,j})}{d\hat{\mathbf{y}}_{:,j}} \Big[\frac{dt_3}{d\hat{\mathbf{y}}}\Big]^{\top} (\mathbf{I} + \hat{\mathbf{K}}\boldsymbol{W})^{-1} (d\boldsymbol{\psi}_1)\Big) \\
&+ \mathrm{Tr}\Big(\boldsymbol{A}^{-1}{\boldsymbol{\psi}_1}^{\top} (\mathbf{I} + \hat{\mathbf{K}}\boldsymbol{W})^{-1\top} \Big[\sum_{i=1}^{n} \mathbf{e}_i \frac{dt_3}{d\hat{\mathbf{y}}_i}\Big]^{\top} \frac{d\log p(\boldsymbol{s}_{:,j}|\hat{\mathbf{y}}_{:,j})}{d\hat{\mathbf{y}}_{:,j}}^{\top} (d\boldsymbol{\psi}_1)\Big) \\
&+ \mathrm{Tr}\Big(\boldsymbol{A}^{-1}{\boldsymbol{\psi}_1}^{\top} \frac{d\log p(\boldsymbol{s}_{:,j}|\hat{\mathbf{y}}_{:,j})}{d\hat{\mathbf{y}}_{:,j}} \Big[\frac{dt_3}{d\hat{\mathbf{y}}}\Big]^{\top} (\mathbf{I} + \hat{\mathbf{K}}\boldsymbol{W})^{-1} \boldsymbol{\psi}_1 \boldsymbol{A}^{-1} \beta {\boldsymbol{\psi}_1}^{\top} (d\boldsymbol{\psi}_1)\Big) \\
&+ \mathrm{Tr}\Big(\beta \boldsymbol{A}^{-1}{\boldsymbol{\psi}_1}^{\top} (\mathbf{I} + \hat{\mathbf{K}}\boldsymbol{W})^{-1\top} \Big[\sum_{i=1}^{n} \mathbf{e}_i^{\top} \frac{dt_3}{d\hat{\mathbf{y}}_i}\Big]^{\top} \frac{d\log p(\boldsymbol{s}_{:,j}|\hat{\mathbf{y}}_{:,j})}{d\hat{\mathbf{y}}_{:,j}}^{\top} \boldsymbol{\psi}_1 \boldsymbol{A}^{-1} {\boldsymbol{\psi}_1}^{\top} (d\boldsymbol{\psi}_1)\Big) \\
&\qquad\qquad\qquad\qquad (\text{use the } A^{\top}B = (B^{\top}A)^{\top}) \\
&= \mathrm{Tr}\Big(\boldsymbol{A}^{-1}{\boldsymbol{\psi}_1}^{\top} \frac{d\log p(\boldsymbol{s}_{:,j}|\hat{\mathbf{y}}_{:,j})}{d\hat{\mathbf{y}}_{:,j}} \Big[\frac{dt_3}{d\hat{\mathbf{y}}}\Big]^{\top} (\mathbf{I} + \hat{\mathbf{K}}\boldsymbol{W})^{-1} (d\boldsymbol{\psi}_1)\Big) \\
&+ \mathrm{Tr}\Big(\boldsymbol{A}^{-1}{\boldsymbol{\psi}_1}^{\top} (\mathbf{I} + \hat{\mathbf{K}}\boldsymbol{W})^{-1\top} \Big[\frac{dt_3}{d\hat{\mathbf{y}}}\Big] \frac{d\log p(\boldsymbol{s}_{:,j}|\hat{\mathbf{y}}_{:,j})}{d\hat{\mathbf{y}}_{:,j}}^{\top} (d\boldsymbol{\psi}_1)\Big) \\
&+ \mathrm{Tr}\Big(\boldsymbol{A}^{-1}{\boldsymbol{\psi}_1}^{\top} \frac{d\log p(\boldsymbol{s}_{:,j}|\hat{\mathbf{y}}_{:,j})}{d\hat{\mathbf{y}}_{:,j}} \Big[\frac{dt_3}{d\hat{\mathbf{y}}}\Big]^{\top} (\mathbf{I} + \hat{\mathbf{K}}\boldsymbol{W})^{-1} \boldsymbol{\psi}_1 \boldsymbol{A}^{-1} \beta {\boldsymbol{\psi}_1}^{\top} (d\boldsymbol{\psi}_1)\Big) \\
&+ \mathrm{Tr}\Big(\beta \boldsymbol{A}^{-1}{\boldsymbol{\psi}_1}^{\top} (\mathbf{I} + \hat{\mathbf{K}}\boldsymbol{W})^{-1\top} \Big[\frac{dt_3}{d\hat{\mathbf{y}}}\Big] \frac{d\log p(\boldsymbol{s}_{:,j}|\hat{\mathbf{y}}_{:,j})}{d\hat{\mathbf{y}}_{:,j}}^{\top} \boldsymbol{\psi}_1 \boldsymbol{A}^{-1} {\boldsymbol{\psi}_1}^{\top} (d\boldsymbol{\psi}_1)\Big) \\
&\qquad\qquad\qquad \Big(F \triangleq \tfrac{d\log p(\boldsymbol{s}_{:,j}|\hat{\mathbf{y}}_{:,j})}{d\hat{\mathbf{y}}_{:,j}} \Big[\tfrac{dt_3}{d\hat{\mathbf{y}}}\Big]^{\top} (\mathbf{I} + \hat{\mathbf{K}}\boldsymbol{W})^{-1}\Big) \\
&= \mathrm{Tr}\Big(\boldsymbol{A}^{-1}{\boldsymbol{\psi}_1}^{\top} (F + F^{\top}) (d\boldsymbol{\psi}_1)\Big) \\
&+ \mathrm{Tr}\Big(\boldsymbol{A}^{-1}{\boldsymbol{\psi}_1}^{\top} (F + F^{\top}) \boldsymbol{\psi}_1 \boldsymbol{A}^{-1} \beta {\boldsymbol{\psi}_1}^{\top} (d\boldsymbol{\psi}_1)\Big)
\end{aligned}$$

For $\mathbf{K}_{\mathbf{uu}}$

$$\begin{aligned}\frac{d\hat{\mathbf{y}}_i}{d\mathbf{K}_{\mathbf{uu}}} &= \mathbf{e}_i^\top(\mathbf{I}+\hat{\mathbf{K}}\boldsymbol{W})^{-1}(d\hat{\mathbf{K}})\frac{d\log p(\boldsymbol{s}_{:,j}|\hat{\mathbf{y}}_{:,j})}{d\hat{\mathbf{y}}_{:,j}}\\ &= \mathbf{e}_i^\top(\mathbf{I}+\hat{\mathbf{K}}\boldsymbol{W})^{-1}(-\boldsymbol{\psi}_1\boldsymbol{A}^{-1}(d\mathbf{K}_{\mathbf{uu}})\boldsymbol{A}^{-1}{\boldsymbol{\psi}_1}^\top)\frac{d\log p(\boldsymbol{s}_{:,j}|\hat{\mathbf{y}}_{:,j})}{d\hat{\mathbf{y}}_{:,j}}\\ &= -\boldsymbol{A}^{-1}{\boldsymbol{\psi}_1}^\top\frac{d\log p(\boldsymbol{s}_{:,j}|\hat{\mathbf{y}}_{:,j})}{d\hat{\mathbf{y}}_{:,j}}\mathbf{e}_i^\top(\mathbf{I}+\hat{\mathbf{K}}\boldsymbol{W})^{-1}\boldsymbol{\psi}_1\boldsymbol{A}^{-1}(d\mathbf{K}_{\mathbf{uu}})\end{aligned}$$

Apply the chain rule

$$\begin{aligned}\frac{dt_3}{d\mathbf{K}_{\mathbf{uu}}} &= \sum_{i=1}^{n}\frac{dt_3}{d\hat{\mathbf{y}}_i}\frac{d\hat{\mathbf{y}}_i}{d\mathbf{K}_{\mathbf{uu}}}\\ &= \mathrm{Tr}\Big(-\boldsymbol{A}^{-1}{\boldsymbol{\psi}_1}^\top\frac{d\log p(\boldsymbol{s}_{:,j}|\hat{\mathbf{y}}_{:,j})}{d\hat{\mathbf{y}}_{:,j}}\Big[\sum_{i=1}^{n}\mathbf{e}_i^\top\frac{dt_3}{d\hat{\mathbf{y}}_i}\Big](\mathbf{I}+\hat{\mathbf{K}}\boldsymbol{W})^{-1}\boldsymbol{\psi}_1\boldsymbol{A}^{-1}(d\mathbf{K}_{\mathbf{uu}})\Big)\\ &= \mathrm{Tr}\Big(-\boldsymbol{A}^{-1}{\boldsymbol{\psi}_1}^\top\frac{d\log p(\boldsymbol{s}_{:,j}|\hat{\mathbf{y}}_{:,j})}{d\hat{\mathbf{y}}_{:,j}}\Big[\frac{dt_3}{d\hat{\mathbf{y}}}\Big]^\top(\mathbf{I}+\hat{\mathbf{K}}\boldsymbol{W})^{-1}\boldsymbol{\psi}_1\boldsymbol{A}^{-1}(d\mathbf{K}_{\mathbf{uu}})\Big)\end{aligned}$$

For β

$$\begin{aligned}\frac{d\hat{\mathbf{y}}_i}{d\beta} &= \mathbf{e}_i^\top(\mathbf{I}+\hat{\mathbf{K}}\boldsymbol{W})^{-1}\frac{d\hat{\mathbf{K}}}{d\beta}\frac{d\log p(\boldsymbol{s}_{:,j}|\hat{\mathbf{y}}_{:,j})}{d\hat{\mathbf{y}}_{:,j}}\\ &= -\mathbf{e}_i^\top(\mathbf{I}+\hat{\mathbf{K}}\boldsymbol{W})^{-1}(\beta^{-2}+\boldsymbol{\psi}_1\boldsymbol{A}^{-1}(\boldsymbol{\psi}_2-{\boldsymbol{\psi}_1}^\top\boldsymbol{\psi}_1)\boldsymbol{A}^{-1}{\boldsymbol{\psi}_1}^\top)(d\beta)\frac{d\log p(\boldsymbol{s}_{:,j}|\hat{\mathbf{y}}_{:,j})}{d\hat{\mathbf{y}}_{:,j}}\\ &= -\frac{d\log p(\boldsymbol{s}_{:,j}|\hat{\mathbf{y}}_{:,j})}{d\hat{\mathbf{y}}_{:,j}}\mathbf{e}_i^\top(\mathbf{I}+\hat{\mathbf{K}}\boldsymbol{W})^{-1}(\beta^{-2}+\boldsymbol{\psi}_1\boldsymbol{A}^{-1}(\boldsymbol{\psi}_2-{\boldsymbol{\psi}_1}^\top\boldsymbol{\psi}_1)\boldsymbol{A}^{-1}{\boldsymbol{\psi}_1}^\top)(d\beta)\end{aligned}$$

Apply the chain rule

$$\begin{aligned}\frac{dt_3}{d\beta} &= \sum_{i=1}^{n}\frac{dt_3}{d\hat{\mathbf{y}}_i}\frac{d\hat{\mathbf{y}}_i}{d\beta}\\ &= \mathrm{Tr}\Big(-\frac{d\log p(\boldsymbol{s}_{:,j}|\hat{\mathbf{y}}_{:,j})}{d\hat{\mathbf{y}}_{:,j}}\Big[\sum_{i=1}^{n}\mathbf{e}_i^\top\frac{dt_3}{d\hat{\mathbf{y}}_i}\Big](\mathbf{I}+\hat{\mathbf{K}}\boldsymbol{W})^{-1}(\beta^{-2}+\boldsymbol{\psi}_1\boldsymbol{A}^{-1}(\boldsymbol{\psi}_2-{\boldsymbol{\psi}_1}^\top\boldsymbol{\psi}_1)\boldsymbol{A}^{-1}{\boldsymbol{\psi}_1}^\top)(d\beta)\Big)\\ &= \mathrm{Tr}\Big(-\frac{d\log p(\boldsymbol{s}_{:,j}|\hat{\mathbf{y}}_{:,j})}{d\hat{\mathbf{y}}_{:,j}}\Big[\frac{dt_3}{d\hat{\mathbf{y}}}\Big]^\top(\mathbf{I}+\hat{\mathbf{K}}\boldsymbol{W})^{-1}(\beta^{-2}+\boldsymbol{\psi}_1\boldsymbol{A}^{-1}(\boldsymbol{\psi}_2-{\boldsymbol{\psi}_1}^\top\boldsymbol{\psi}_1)\boldsymbol{A}^{-1}{\boldsymbol{\psi}_1}^\top)(d\beta)\Big)\end{aligned}$$

$$\frac{dt_3}{d\psi_0} = 0$$

t_4 derivatives

$$\frac{dt_4}{d\boldsymbol{\theta}_K}_{\text{explicit}} = \frac{d(-\frac{p}{2}\log|\boldsymbol{A}|)}{d\boldsymbol{\theta}_K} = -\frac{p}{2}\mathrm{Tr}\Big(\boldsymbol{A}^{-1}\frac{d\boldsymbol{A}}{d\boldsymbol{\theta}_K}\Big)$$

t_5 derivatives

$$\frac{dt_5}{d\boldsymbol{\theta}_K}_{\text{explicit}} = \frac{d(-\frac{p}{2}\log|\mathbf{K}_{\mathbf{uu}}^{-1}|)}{d\boldsymbol{\theta}_K} = \frac{d(\frac{p}{2}\log|\mathbf{K}_{\mathbf{uu}}|)}{d\boldsymbol{\theta}_K} = \frac{p}{2}\mathrm{Tr}\Big(\mathbf{K}_{\mathbf{uu}}^{-1}\frac{d\mathbf{K}_{\mathbf{uu}}}{d\boldsymbol{\theta}_K}\Big)$$

E.2 Computational Burden

The marginal likelihood and derivatives (Section E.1.1) require the following terms, $(\hat{\mathbf{K}}^{-1}+\boldsymbol{W})^{-1}$, $(\hat{\mathbf{K}}+\boldsymbol{W}^{-1})^{-1}$. As in Rasmussen and Williams (2006) these can be formulated in terms a matrix

$$\boldsymbol{B} = \mathbf{I} + \boldsymbol{W}^{\frac{1}{2}}\hat{\mathbf{K}}\boldsymbol{W}^{\frac{1}{2}} \tag{E.3}$$

in the following ways

$$\begin{aligned}
(\hat{\mathbf{K}}+\boldsymbol{W}^{-1})^{-1} &= \boldsymbol{W}^{\frac{1}{2}}\boldsymbol{W}^{-\frac{1}{2}}(\hat{\mathbf{K}}+\boldsymbol{W}^{-1})^{-1}\boldsymbol{W}^{-\frac{1}{2}}\boldsymbol{W}^{\frac{1}{2}}\\
&= \boldsymbol{W}^{\frac{1}{2}}(\mathbf{I}+\boldsymbol{W}^{\frac{1}{2}}\hat{\mathbf{K}}\boldsymbol{W}^{\frac{1}{2}})^{-1}\boldsymbol{W}^{\frac{1}{2}}\\
&= \boldsymbol{W}^{\frac{1}{2}}\boldsymbol{B}^{-1}\boldsymbol{W}^{\frac{1}{2}}
\end{aligned}$$

and

$$\begin{aligned}
(\hat{\mathbf{K}}^{-1}+\boldsymbol{W})^{-1} &= (\hat{\mathbf{K}}^{-1}+\boldsymbol{W}^{\frac{1}{2}}\mathbf{I}\boldsymbol{W}^{\frac{1}{2}})^{-1}\\
&= \hat{\mathbf{K}} - \hat{\mathbf{K}}\boldsymbol{W}^{\frac{1}{2}}(\mathbf{I}+\boldsymbol{W}^{\frac{1}{2}}\hat{\mathbf{K}}\boldsymbol{W}^{\frac{1}{2}})^{-1}\boldsymbol{W}^{\frac{1}{2}}\hat{\mathbf{K}}\\
&= \hat{\mathbf{K}} - \hat{\mathbf{K}}\boldsymbol{W}^{\frac{1}{2}}\boldsymbol{B}^{-1}\boldsymbol{W}^{\frac{1}{2}}\hat{\mathbf{K}}
\end{aligned}$$

However although stable to invert due to the addition of a positive diagonal $\mathbf{I}$, in its naive form, $\boldsymbol{B}^{-1}$ requires an $\mathcal{O}(n^3)$ inversion, as $\boldsymbol{B} \in \mathbb{R}^{n\times n}$, which is wasteful. However the particular form of $\hat{\mathbf{K}}$ allows us to reduce this computational burden. From

Equation (5.22),

$$\begin{aligned}\hat{\mathbf{K}}^{-1} &= \beta^{-1}\mathbf{I} + \boldsymbol{\psi}_1(\underbrace{\mathbf{K}_{\mathbf{uu}} + \beta(\boldsymbol{\psi}_2 - \boldsymbol{\psi}_1{}^\top\boldsymbol{\psi}_1)}_{\boldsymbol{A}})^{-1}\boldsymbol{\psi}_1{}^\top \\ &= \beta^{-1}\mathbf{I} + \boldsymbol{\psi}_1\boldsymbol{A}^{-1}\boldsymbol{\psi}_1{}^\top\end{aligned}$$

We can then rewrite the key equation for $\boldsymbol{B}^{-1}$ using the Matrix Inversion Lemma (Section A.2.1)

$$\begin{aligned}\boldsymbol{B}^{-1} &= (\mathbf{I} + \boldsymbol{W}^{\frac{1}{2}}\hat{\mathbf{K}}\boldsymbol{W}^{\frac{1}{2}})^{-1} \\ &= (\mathbf{I} + \boldsymbol{W}^{\frac{1}{2}}(\beta^{-1}\mathbf{I} + \boldsymbol{\psi}_1\boldsymbol{A}\boldsymbol{\psi}_1)^{-1}\boldsymbol{\psi}_1)^{-1} \\ &= (\underbrace{\mathbf{I} + \boldsymbol{W}^{\frac{1}{2}}\beta^{-1}\mathbf{I}\boldsymbol{W}^{\frac{1}{2}}}_{\boldsymbol{Q}} + \boldsymbol{W}^{\frac{1}{2}}\boldsymbol{\psi}_1\boldsymbol{A}^{-1}\boldsymbol{\psi}_1{}^\top\boldsymbol{W}^{\frac{1}{2}})^{-1} \\ &= (\underbrace{\boldsymbol{Q}}_{A} + \underbrace{\boldsymbol{W}^{\frac{1}{2}}\boldsymbol{\psi}_1}_{U}\underbrace{\boldsymbol{A}^{-1}}_{C}\underbrace{\boldsymbol{\psi}_1{}^\top\boldsymbol{W}^{\frac{1}{2}}}_{V})^{-1} \\ &= A^{-1} - A^{-1}U(C^{-1} + VA^{-1}U)^{-1}VA^{-1} \\ &= \boldsymbol{Q}^{-1} - \boldsymbol{Q}^{-1}\boldsymbol{W}^{\frac{1}{2}}\boldsymbol{\psi}_1(\boldsymbol{A} + \boldsymbol{\psi}_1{}^\top\boldsymbol{W}^{\frac{1}{2}}\boldsymbol{Q}^{-1}\boldsymbol{W}^{\frac{1}{2}}\boldsymbol{\psi}_1)^{-1}\boldsymbol{\psi}_1{}^\top\boldsymbol{W}^{\frac{1}{2}}\boldsymbol{Q}^{-1}.\end{aligned}$$

If $\boldsymbol{W}^{\frac{1}{2}}$ is diagonal, which is the case for factorising likelihoods, such as the log-logistic, the computational burden is linear to invert $\boldsymbol{Q}$. Depending on the number of data n, the inversion of $\boldsymbol{A} + \boldsymbol{\psi}_1{}^\top\boldsymbol{W}^{\frac{1}{2}}\boldsymbol{Q}^{-1}\boldsymbol{W}^{\frac{1}{2}}\boldsymbol{\psi}_1 \in \mathbb{R}^{m\times m}$, or the multiplication of $\boldsymbol{Q}^{-1}\boldsymbol{W}^{\frac{1}{2}}\boldsymbol{\psi}_1$ will be the dominating factors, neither are $\mathcal{O}(n^3)$.

E.3 Posterior and Effective Likelihood

The Laplace approximation, Section 3.2.1, used for the non-Gaussian likelihood BGPLVM introduced in Section 5.6; the Gaussian posterior for $\mathbf{y}_{:,j}$ was found to be,

$$\mathcal{N}\left(\mathbf{y}_{:,j}|\hat{\mathbf{y}}_{:,j}, (\hat{\mathbf{K}}^{-1} + \boldsymbol{W})^{-1}\right). \tag{E.4}$$

The prior used was $p(\mathbf{y}_{:,j}) = \mathcal{N}\left(\mathbf{y}_{:,j}|\mathbf{0}, \hat{\mathbf{K}}\right)$, where $\hat{\mathbf{K}}$ is as in Equation (5.22).

Since the approximate posterior $p(\mathbf{y}_{:,j}|\boldsymbol{s}_{:,j})$ is Gaussian, and the prior $p(\mathbf{y}_{:,j})$ is Gaussian, it suggests that there some be some *effective likelihood*, that is Gaussian with respect to some *effective data*, $\tilde{\boldsymbol{s}}$, with *effective* covariance $\tilde{\boldsymbol{\Sigma}}$, such that the posterior matches Equation (E.4),

$$\begin{aligned}
p(\mathbf{y}_{:,j}|\boldsymbol{s}_{:,j}) &= \frac{p(\boldsymbol{s}_{:,j}|\mathbf{y}_{:,j})p(\mathbf{y}_{:,j})}{p(\boldsymbol{s}_{:,j})} && \text{(E.5)}\\
\mathcal{N}\left(\mathbf{y}_{:,j}|\hat{\mathbf{y}}_{:,j},(\hat{\mathbf{K}}^{-1}+\boldsymbol{W})^{-1}\right) &\propto p(\boldsymbol{s}_{:,j}|\mathbf{y}_{:,j})\mathcal{N}\left(\mathbf{y}_{:,j}|\mathbf{0},\hat{\mathbf{K}}\right) && \text{(E.6)}\\
\mathcal{N}\left(\mathbf{y}_{:,j}|\hat{\mathbf{y}}_{:,j},(\hat{\mathbf{K}}^{-1}+\boldsymbol{W})^{-1}\right) &\propto \mathcal{N}\left(\tilde{\boldsymbol{s}}_{:,j}|\mathbf{y}_{:,j},\tilde{\boldsymbol{\Sigma}}\right)\mathcal{N}\left(\mathbf{y}_{:,j}|\mathbf{0},\hat{\mathbf{K}}\right) && \text{(E.7)}
\end{aligned}$$

To find the associated effective data, and covariance of this effective likelihood, begin by rearranging the above equation and ignoring constant terms in $\mathbf{y}$,

$$\begin{aligned}
\mathcal{N}\left(\tilde{\boldsymbol{s}}_{:,j}|\mathbf{y}_{:,j},\tilde{\boldsymbol{\Sigma}}\right) &\propto \frac{\mathcal{N}\left(\mathbf{y}_{:,j}|\hat{\mathbf{y}}_{:,j},(\hat{\mathbf{K}}^{-1}+\boldsymbol{W})^{-1}\right)}{\mathcal{N}\left(\mathbf{y}_{:,j}|\mathbf{0},\hat{\mathbf{K}}\right)}\\
\exp((\tilde{\boldsymbol{s}}_{:,j}-\mathbf{y}_{:,j})^\top\tilde{\boldsymbol{\Sigma}}^{-1}(\tilde{\boldsymbol{s}}_{:,j}-\mathbf{y}_{:,j})) &\propto \exp((\mathbf{y}_{:,j}-\hat{\mathbf{y}}_{:,j})^\top(\hat{\mathbf{K}}^{-1}+\boldsymbol{W})(\mathbf{y}_{:,j}-\hat{\mathbf{y}}_{:,j}))\\
&\quad\times\exp(-\mathbf{y}_{:,j}^\top\hat{\mathbf{K}}^{-1}\mathbf{y}_{:,j})\\
\exp((\tilde{\boldsymbol{s}}_{:,j}-\mathbf{y}_{:,j})^\top\tilde{\boldsymbol{\Sigma}}^{-1}(\tilde{\boldsymbol{s}}_{:,j}-\mathbf{y}_{:,j})) &\propto \exp(\mathbf{y}_{:,j}^\top\boldsymbol{W}\mathbf{y}_{:,j}+2\mathbf{y}_{:,j}^\top(\hat{\mathbf{K}}^{-1}+\boldsymbol{W})\hat{\mathbf{y}}_{:,j})
\end{aligned}$$

The effective covariance and data can then be found by considering the linear and quadratic terms required to make the right hand side match the form of the Gaussian on the left,

$$\begin{aligned}
\tilde{\boldsymbol{\Sigma}} &= \boldsymbol{W}^{-1} && \text{(E.8)}\\
\mathbf{y}_{:,j}^\top\tilde{\boldsymbol{\Sigma}}^{-1}\tilde{\boldsymbol{s}}_{:,j} &= \mathbf{y}_{:,j}^\top(\hat{\mathbf{K}}^{-1}+\boldsymbol{W})\hat{\mathbf{y}}_{:,j}\\
\boldsymbol{s}_{:,j} &= \tilde{\boldsymbol{\Sigma}}(\hat{\mathbf{K}}^{-1}+\boldsymbol{W})\hat{\mathbf{y}}_{:,j}\\
&= \boldsymbol{W}^{-1}(\hat{\mathbf{K}}^{-1}+\boldsymbol{W})\hat{\mathbf{y}}_{:,j} && \text{(E.9)}\\
&= \boldsymbol{W}^{-1}\hat{\mathbf{K}}^{-1}\hat{\mathbf{y}}_{:,j}+\boldsymbol{W}^{-1}\boldsymbol{W}\hat{\mathbf{y}}_{:,j}\\
&= \boldsymbol{W}^{-1}\hat{\mathbf{K}}^{-1}\hat{\mathbf{K}}\frac{d\log p(\boldsymbol{s}_{:,j}|\hat{\mathbf{y}}_{:,j})}{d\hat{\mathbf{y}}_{:,j}}+\hat{\mathbf{y}}_{:,j}\\
&= \boldsymbol{W}^{-1}\frac{d\log p(\boldsymbol{s}_{:,j}|\hat{\mathbf{y}}_{:,j})}{d\hat{\mathbf{y}}_{:,j}}+\hat{\mathbf{y}}_{:,j} && \text{(E.10)}
\end{aligned}$$

Note that each of these quantities have been computed during the Laplace approximation. We will use Equation (E.9) in practice. However it is interesting to note that this same effective likelihood, written as in Equation (E.10), was previously used by Vehtari et al. (2014) in order to compute an approximation leave-one-out error for the Laplace approximation.

The effective Gaussian likelihood, that gives the same posterior as the Laplace approximation using the non-Gaussian likelihood, is then,

$$\mathcal{N}\left(\tilde{\boldsymbol{s}}_{:,j}|\mathbf{y}_{:,j},\boldsymbol{W}^{-1}\right) \tag{E.11}$$

This could be also used for evaluating the approximate LOO error of the model.

For sparse GP prediction, it is useful to have the low-rank posterior, $p(\mathbf{u}_{:,j}|\boldsymbol{s}_{:,j})$,

$$p(\mathbf{u}_{:,j}|\boldsymbol{s}_{:,j}) = \int p(\mathbf{u}_{:,j}|\mathbf{y}_{:,j})p(\mathbf{y}_{:,j}|\boldsymbol{s}_{:,j})d\mathbf{y}_{:,j}$$

From the derivation of the BGPLVM, we found that the optimal form for $q(\mathbf{u}_{:,j}|\boldsymbol{\theta}_V)$ was,

$$\begin{aligned} q(\mathbf{u}_{:,j}|\boldsymbol{\theta}_V) &= \mathcal{N}\left(\mathbf{u}_{:,j}|\mathbf{m}_{:,j},\mathbf{R}\right) \\ \mathbf{m}_{:,j} &= \mathbf{R}\mathbf{K}_{\mathbf{uu}}^{-1}\boldsymbol{\psi}_1{}^{\top}\beta\mathbf{y}_{:,j} \\ \mathbf{R} &= (\mathbf{K}_{\mathbf{uu}}^{-1}\beta\boldsymbol{\psi}_2\mathbf{K}_{\mathbf{uu}}^{-1}+\mathbf{K}_{\mathbf{uu}}^{-1})^{-1} \end{aligned}$$

This is the optimal approximation for the prior used for the LABGPLVM, $p(\mathbf{y}_{:,j})$, which is a BGPLVM bound on the marginal likelihood. Note however that for the LABGPLVM, $\mathbf{y}_{:,j}$, is latent, and so we wish to marginalise out the possible functions it may take.

We have also found through the Laplace approximation that the posterior $p(\mathbf{y}_{:,j}|\boldsymbol{s}_{:,j})$ is,

$$p(\mathbf{y}_{:,j}|\boldsymbol{s}_{:,j}) = \mathcal{N}\left(\mathbf{y}_{:,j}|\hat{\mathbf{y}}_{:,j},(\hat{\mathbf{K}}^{-1}+\boldsymbol{W})^{-1}\right)$$

The low rank approximate posterior for the LABGPLVM model can be found simply by marginalising our $\mathbf{y}_{:,j}$ using the Gaussian identities for marginalisation, Section A.1.1,

$$q(\mathbf{u}_{:,j}|\boldsymbol{\theta}_V) = \mathcal{N}\left(\mathbf{u}_{:,j}|\mathbf{R}\mathbf{K}_{\mathbf{uu}}^{-1}\boldsymbol{\psi}_1{}^{\top}\beta_j\hat{\mathbf{y}}_{:,j},\mathbf{R}+\mathbf{R}\mathbf{K}_{\mathbf{uu}}^{-1}\boldsymbol{\psi}_1{}^{\top}\beta_j(\hat{\mathbf{K}}^{-1}+\boldsymbol{W}_j)^{-1}\beta_j\boldsymbol{\psi}_1\mathbf{K}_{\mathbf{uu}}^{-1}\mathbf{R}\right)$$

This can be further expanded to,

$$
\begin{aligned}
q(\mathbf{u}_{:,j}|\boldsymbol{\theta}_V) &= \mathcal{N}\left(\mathbf{u}_{:,j}|\boldsymbol{\mu}_V, \boldsymbol{\Sigma}_V\right) && \text{(E.12)}\\
\boldsymbol{\mu}_V &= \mathbf{K}_{\mathbf{uu}}\boldsymbol{H}^{-1}{\boldsymbol{\psi}_1}^{\top}\beta_j\hat{\mathbf{y}}_{:,j} && \text{(E.13)}\\
\boldsymbol{\Sigma}_V &= \mathbf{K}_{\mathbf{uu}}\left[\boldsymbol{H}^{-1} + \boldsymbol{H}^{-1}{\boldsymbol{\psi}_1}^{\top}\beta_j\left(\hat{\mathbf{K}}^{-1} + \boldsymbol{W}_j\right)^{-1}\beta_j\boldsymbol{\psi}_1\boldsymbol{H}^{-1}\right]\mathbf{K}_{\mathbf{uu}} && \text{(E.14)}\\
\boldsymbol{H} &\triangleq \left(\mathbf{K}_{\mathbf{uu}} + \beta_j\boldsymbol{\psi}_2\right) && \text{(E.15)}
\end{aligned}
$$

Again, as a result of the form or $(\hat{\mathbf{K}}^{-1} + \boldsymbol{W}_j)^{-1}$, this only requires $\mathcal{O}(nm^2)$ to compute; and so subsequent posterior predictions share this computational complexity.

This distribution can be subsequently used for prediction, at new test points, $\mathbf{X}^*$.

E.4 Uncertain Input Prediction

As discussed in Section 6.1 when the input is uncertain for a Gaussian process, the posterior prediction no longer has a Gaussian form. A commonly used approach is to approximate the distribution with a Gaussian distribution. Following Deisenroth (2010) derivation, we can compute the moments of the non-Gaussian predictive distribution, when the inputs are uncertain, through the law of iterated expectations.

$$
\begin{aligned}
p(\boldsymbol{f}^*|\mathbf{y}) &= \int p(\boldsymbol{f}^*|\mathbf{y}, \mathbf{X}^*)p(\mathbf{X}^*)d\mathbf{x}^*\\
&\approx \mathcal{N}\left(\boldsymbol{f}^*|m(\boldsymbol{\mu}, \boldsymbol{S}), v(\boldsymbol{\mu}, \boldsymbol{S})\right)
\end{aligned}
$$

where $p(\boldsymbol{f}^*|\mathbf{y}, \mathbf{X}^*)$ is the predictive distribution of a Gaussian process (be that a sparse Gaussian process, or a normal Gaussian process), with known inputs. $p(\mathbf{X}^*)$ is the uncertainty about $\mathbf{X}^*$ and $\boldsymbol{\mu}$ and $\boldsymbol{S}$ are the mean and variance of the uncertain input. For the BGPLVM, $p(\mathbf{X}^*)$ will typically be the posterior distribution, for the uncertainty over our test points, $q(\mathbf{X}^*|\boldsymbol{\theta}_V)$. In what follows we will derive the distribution assuming we have uncertain inputs, $q(\mathbf{X}^*|\boldsymbol{\theta}_V)$, and so $\boldsymbol{\mu} = \boldsymbol{\mu}_V^*$, and $\boldsymbol{S} = \boldsymbol{\Sigma}_V^*$, as would be the case when making predictions for the BGPLVM typically. We will focus on a single univariate prediction, $\mathbf{x}^*$, treating each input seperately. Then $q(\mathbf{x}^*|\boldsymbol{\mu}_V^*, \boldsymbol{\Sigma}_V^*)$ is a marginal of $q(\mathbf{X}^*|\boldsymbol{\theta}_V)$. $q(\mathbf{X}^*|\boldsymbol{\theta}_V)$ is usually a made with a mean field approximation, $\boldsymbol{\Sigma}_V^*$ will be diagonal, and so considering the univariate case is appropriate. The multivariate version may also be derived simiarly, see Candela et al. (2003); Deisenroth (2010); Girard et al. (2003) for details. The mean and variance of the approximate predictive distribution are as follows,

$$m(\boldsymbol{\mu}_V^*, \boldsymbol{\Sigma}_V^*) = \mathbb{E}_{q(\mathbf{x}^*|\boldsymbol{\mu}_V^*,\boldsymbol{\Sigma}_V^*)}\left[\mathbb{E}_{p(\boldsymbol{f}^*|\mathbf{y})}\left[\boldsymbol{f}^*|\mathbf{x}^*\right]\right] = \mathbb{E}_{q(\mathbf{x}^*|\boldsymbol{\mu}_V^*,\boldsymbol{\Sigma}_V^*)}\left[\mu(\mathbf{x}^*)\right] \tag{E.16}$$

$$\begin{aligned} v(\boldsymbol{\mu}_V^*, \boldsymbol{\Sigma}_V^*) &= \mathbb{E}_{q(\mathbf{x}^*|\boldsymbol{\mu}_V^*,\boldsymbol{\Sigma}_V^*)}\left[\mathrm{var}_{p(\boldsymbol{f}^*|\mathbf{y})}[\boldsymbol{f}^*|\mathbf{x}^*]\right] + \mathrm{var}_{q(\mathbf{x}^*|\boldsymbol{\mu}_V^*,\boldsymbol{\Sigma}_V^*)}[\mathbb{E}_{p(\boldsymbol{f}^*|\mathbf{Y})}\left[\boldsymbol{f}^*|\mathbf{x}^*\right]] \\ &= \mathbb{E}_{q(\mathbf{x}^*|\boldsymbol{\mu}_V^*,\boldsymbol{\Sigma}_V^*)}\left[\sigma^2(\mathbf{x}^*)\right] + \mathrm{var}_{q(\mathbf{x}^*|\boldsymbol{\mu}_V^*,\boldsymbol{\Sigma}_V^*)}[\mu(\mathbf{x}^*)] \\ &= \mathbb{E}_{q(\mathbf{x}^*|\boldsymbol{\mu}_V^*,\boldsymbol{\Sigma}_V^*)}\left[\sigma^2(\mathbf{x}^*)\right] + \mathbb{E}_{q(\mathbf{x}^*|\boldsymbol{\mu}_V^*,\boldsymbol{\Sigma}_V^*)}\left[\mu(\mathbf{x}^*)^2\right] - \mathbb{E}_{q(\mathbf{x}^*|\boldsymbol{\mu}_V^*,\boldsymbol{\Sigma}_V^*)}\left[\mu(\mathbf{x}^*)\right]^2 \end{aligned} \tag{E.17}$$

where $\mu(\mathbf{x}^*)$ and $\sigma^2(\mathbf{x}^*)$ are the mean and variance of the posterior predictive distribution $p(\boldsymbol{f}^*|\mathbf{y})$ respectively,

For clarity we will use the notation $\mathbf{K}_{\boldsymbol{f}^*\boldsymbol{f}^*} = k(\mathbf{x}^*, \mathbf{x}^*)$, $\mathbf{K}_{\boldsymbol{f}\boldsymbol{f}^*} = k(\mathbf{X}, \mathbf{x}^*)$, $\mathbf{K}_{\boldsymbol{f}^*\boldsymbol{f}} = k(\mathbf{x}^*, \mathbf{X})$, $\mathbf{K}_{\mathbf{uu}} = k(\mathbf{Z}, \mathbf{Z})$, $\mathbf{K}_{\mathbf{u}\boldsymbol{f}} = k(\mathbf{Z}, \mathbf{X})$, $\mathbf{K}_{\boldsymbol{f}\mathbf{u}} = k(\mathbf{X}, \mathbf{Z})$, $\mathbf{K}_{\boldsymbol{f}^*\mathbf{u}} = k(\mathbf{x}^*, \mathbf{Z})$, $\mathbf{K}_{\mathbf{u}\boldsymbol{f}^*} = k(\mathbf{Z}, \mathbf{x}^*)$, and as well as equations for ψ_0, $\boldsymbol{\psi}_1$, $\boldsymbol{\psi}_2$, Equations (5.13), (5.14), (5.15) respectively. The class of Gaussian process models covered in this thesis have a range of posterior predictive distributions, each of which share a similar pattern to one another.

For a standard Gaussian process the posterior predictive distribution is a Gaussian with a mean and variance as follows:

$$\begin{aligned} \mu(\mathbf{x}^*) &= \mathbf{K}_{\boldsymbol{f}^*\boldsymbol{f}} \underbrace{\left(\mathbf{K}_{\boldsymbol{f}\boldsymbol{f}} + \beta^{-1}\mathbf{I}\right)^{-1}\mathbf{y}}_{\boldsymbol{D}} \\ \sigma^2(\mathbf{x}^*) &= \mathbf{K}_{\boldsymbol{f}^*\boldsymbol{f}^*} - \mathbf{K}_{\boldsymbol{f}^*\boldsymbol{f}} \underbrace{\left(\mathbf{K}_{\boldsymbol{f}\boldsymbol{f}} + \beta^{-1}\mathbf{I}\right)^{-1}}_{\boldsymbol{E}} \mathbf{K}_{\boldsymbol{f}\boldsymbol{f}^*} \end{aligned}$$

For a sparse Gaussian processes, as in Equation (3.18), the form is as follows:

$$\begin{aligned} \mu(\mathbf{x}^*) &= \mathbf{K}_{\boldsymbol{f}^*\mathbf{u}} \underbrace{\mathbf{K}_{\mathbf{uu}}^{-1}\boldsymbol{\mu}_V}_{\boldsymbol{D}} \\ \sigma^2(\mathbf{x}^*) &= \mathbf{K}_{\boldsymbol{f}^*\boldsymbol{f}^*} - \mathbf{K}_{\boldsymbol{f}^*\mathbf{u}} \underbrace{\left(\mathbf{K}_{\mathbf{uu}}^{-1} - \mathbf{K}_{\mathbf{uu}}^{-1}\boldsymbol{\Sigma}_V\mathbf{K}_{\mathbf{uu}}^{-1}\right)}_{\boldsymbol{E}} \mathbf{K}_{\mathbf{u}\boldsymbol{f}^*} \end{aligned}$$

For a standard sparse Gaussian process with certain training inputs (Section 3.3), $\mathbf{X}$, this results in:

$$\begin{aligned}
\mu(\mathbf{x}^*) &= \mathbf{K}_{\boldsymbol{f}^*\mathbf{u}} \underbrace{\beta(\mathbf{K}_{\mathbf{uu}} + \beta\mathbf{K}_{\mathbf{u}\boldsymbol{f}}\mathbf{K}_{\boldsymbol{f}\mathbf{u}})^{-1}\mathbf{K}_{\mathbf{u}\boldsymbol{f}}\mathbf{Y}}_{\boldsymbol{D}} \\
\sigma^2(\mathbf{x}^*) &= \mathbf{K}_{\boldsymbol{f}^*\boldsymbol{f}^*} - \mathbf{K}_{\boldsymbol{f}^*\mathbf{u}} \underbrace{(\mathbf{K}_{\mathbf{uu}}^{-1} - (\mathbf{K}_{\mathbf{uu}} + \beta\mathbf{K}_{\mathbf{u}\boldsymbol{f}}\mathbf{K}_{\boldsymbol{f}\mathbf{u}})^{-1})}_{\boldsymbol{E}} \mathbf{K}_{\mathbf{u}\boldsymbol{f}^*}
\end{aligned}$$

For the BGPLVM (with uncertain training inputs):

$$\begin{aligned}
\mu(\mathbf{x}^*) &= \mathbf{K}_{\boldsymbol{f}^*\mathbf{u}} \underbrace{\beta(\beta\boldsymbol{\psi}_2 + \mathbf{K}_{\mathbf{uu}})^{-1}{\boldsymbol{\psi}_1}^{\top}\mathbf{y}}_{\boldsymbol{D}} \\
\sigma^2(\mathbf{x}^*) &= \mathbf{K}_{\boldsymbol{f}^*\boldsymbol{f}^*} - \mathbf{K}_{\boldsymbol{f}^*\mathbf{u}} \underbrace{(\mathbf{K}_{\mathbf{uu}}^{-1} - (\beta\boldsymbol{\psi}_2 + \mathbf{K}_{\mathbf{uu}})^{-1})}_{\boldsymbol{E}} \mathbf{K}_{\mathbf{u}\boldsymbol{f}^*}
\end{aligned}$$

For the non-Gaussian likelihood BGPLVM (with uncertain training inputs), using Equations (E.14) and (E.13):

$$\begin{aligned}
\mu(\mathbf{x}^*) &= \mathbf{K}_{\boldsymbol{f}^*\mathbf{u}} \underbrace{\boldsymbol{H}^{-1}{\boldsymbol{\psi}_1}^{\top}\beta_j\hat{\mathbf{y}}_{:,j}}_{\boldsymbol{D}} \\
\sigma^2(\mathbf{x}^*) &= \mathbf{K}_{\boldsymbol{f}^*\boldsymbol{f}^*} - \mathbf{K}_{\boldsymbol{f}^*\mathbf{u}} \underbrace{(\mathbf{K}_{\mathbf{uu}}^{-1} - (\boldsymbol{H}^{-1} + \boldsymbol{H}^{-1}{\boldsymbol{\psi}_1}^{\top}\beta_j(\hat{\mathbf{K}}^{-1} + \boldsymbol{W}_j)^{-1}\beta_j\boldsymbol{\psi}_1\boldsymbol{H}^{-1}))}_{\boldsymbol{E}} \mathbf{K}_{\mathbf{u}\boldsymbol{f}^*} \\
\boldsymbol{H} &\triangleq (\mathbf{K}_{\mathbf{uu}} + \beta_j\boldsymbol{\psi}_2)
\end{aligned}$$

Note that in the case of the standard Gaussian process the predictive mean and variance have the form,

$$\begin{aligned}
\mu(\mathbf{x}^*) &= \mathbf{K}_{\boldsymbol{f}^*\boldsymbol{f}}\boldsymbol{D} \\
\sigma^2(\mathbf{x}^*) &= \mathbf{K}_{\boldsymbol{f}^*\boldsymbol{f}^*} - \mathbf{K}_{\boldsymbol{f}^*\boldsymbol{f}}\boldsymbol{E}\mathbf{K}_{\boldsymbol{f}\boldsymbol{f}^*},
\end{aligned}$$

and the sparse Gaussian process based methods take the form,

$$\mu(\mathbf{x}^*) = \mathbf{K}_{\boldsymbol{f}^*\mathbf{u}}\boldsymbol{D} \tag{E.18}$$

$$\sigma^2(\mathbf{x}^*) = \mathbf{K}_{\boldsymbol{f}^*\boldsymbol{f}^*} - \mathbf{K}_{\boldsymbol{f}^*\mathbf{u}}\boldsymbol{E}\mathbf{K}_{\mathbf{u}\boldsymbol{f}^*}. \tag{E.19}$$

We can use Equation (E.18) and Equations (E.16), (E.17), in order to compute the moments of the predictive mean and variance when the input $\mathbf{x}^*$ contains uncertainty itself. In this case the uncertainty in the input must be integrated out as follows,

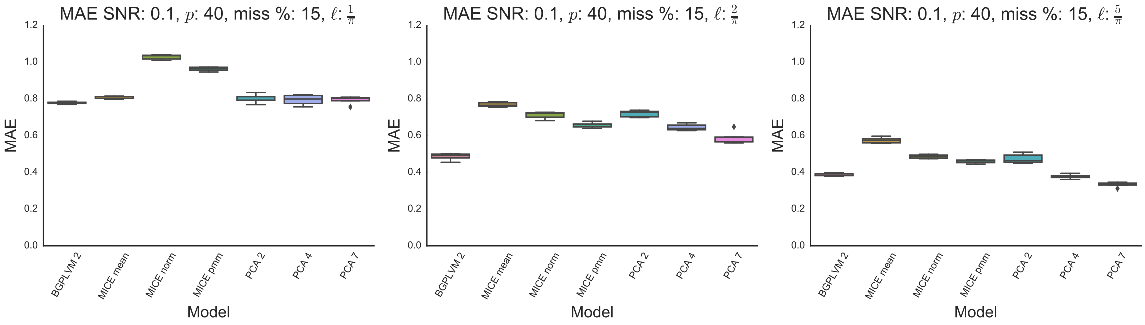
$$
\begin{aligned}
m(\boldsymbol{\mu}_V^*, \boldsymbol{\Sigma}_V^*) &= \int \mu(\mathbf{x}^*) q(\mathbf{x}^* | \boldsymbol{\mu}_V^*, \boldsymbol{\Sigma}_V^*) d\mathbf{x} \\
&= \boldsymbol{D}^\top \underbrace{\int \mathbf{K}(\mathbf{X}, \mathbf{x}^*) q(\mathbf{x}^* | \boldsymbol{\mu}_V^*, \boldsymbol{\Sigma}_V^*)}_{\boldsymbol{\psi}_1{}^*} \\
&= \boldsymbol{D}^\top \boldsymbol{\psi}_1{}^* && \text{(E.20)} \\
v(\boldsymbol{\mu}_V^*, \boldsymbol{\Sigma}_V^*) &= \int \underbrace{\mathbf{K}(\mathbf{x}^*, \mathbf{x}^*)}_{(1\times 1)} - \mathbf{K}(\mathbf{x}^*, \mathbf{X}) \boldsymbol{E} \mathbf{K}(\mathbf{X}, \mathbf{x}^*) q(\mathbf{x}^* | \boldsymbol{\mu}_V^*, \boldsymbol{\Sigma}_V^*) d\mathbf{x}^* \\
&\quad + \int \underbrace{\mathbf{K}(\mathbf{x}^*, \mathbf{X})}_{1\times n} \boldsymbol{D}\boldsymbol{D}^\top \underbrace{\mathbf{K}(\mathbf{X}, \mathbf{x}^*)}_{n\times 1} q(\mathbf{x}^* | \boldsymbol{\mu}_V^*, \boldsymbol{\Sigma}_V^*) d\mathbf{x}^* - \boldsymbol{D}\boldsymbol{\psi}_1{}^* \boldsymbol{\psi}_1{}^{*\top} \boldsymbol{D}^\top \\
&= \int \psi_0{}^* - \operatorname{tr}\left(\boldsymbol{E} \underbrace{\left[\int \mathbf{K}(\mathbf{X}, \mathbf{x}^*) \mathbf{K}(\mathbf{x}^*, \mathbf{X}) q(\mathbf{x}^* | \boldsymbol{\mu}_V^*, \boldsymbol{\Sigma}_V^*) d\mathbf{x}^* \right]}_{\boldsymbol{\psi}_2{}^*} \right) \\
&\quad + \int \boldsymbol{D}^\top \underbrace{\left[\mathbf{K}(\mathbf{X}, \mathbf{x}^*) \mathbf{K}(\mathbf{x}^*, \mathbf{X}) q(\mathbf{x}^* | \boldsymbol{\mu}_V^*, \boldsymbol{\Sigma}_V^*) d\mathbf{x}^* \right]}_{\boldsymbol{\psi}_2{}^*} \boldsymbol{D} \\
&\quad - \boldsymbol{D}\boldsymbol{\psi}_1{}^* \boldsymbol{\psi}_1{}^{*\top} \boldsymbol{D}^\top \\
&= \psi_0{}^* - \operatorname{tr}\left(\boldsymbol{E} \boldsymbol{\psi}_2{}^* \right) + \boldsymbol{D}^\top (\boldsymbol{\psi}_2{}^* - \boldsymbol{\psi}_1{}^* \boldsymbol{\psi}_1{}^{*\top}) \boldsymbol{D} && \text{(E.21)}
\end{aligned}
$$

where $\psi_0{}^* = \int \mathbf{K}(\mathbf{x}^*, \mathbf{x}^*) q(\mathbf{x}^* | \boldsymbol{\mu}_V^*, \boldsymbol{\Sigma}_V^*) d\mathbf{x}^* \in \mathbb{R}^{1\times 1}$, $\boldsymbol{\psi}_1{}^* = \int \mathbf{K}(\mathbf{X}, \mathbf{x}^*) q(\mathbf{x}^* | \boldsymbol{\mu}_V^*, \boldsymbol{\Sigma}_V^*) \in \mathbb{R}^{n\times 1}$, and $\boldsymbol{\psi}_2{}^* = \int \mathbf{K}(\mathbf{X}, \mathbf{x}^*) \mathbf{K}(\mathbf{x}^*, \mathbf{X}) q(\mathbf{x}^* | \boldsymbol{\mu}_V^*, \boldsymbol{\Sigma}_V^*) \in \mathbb{R}^{1\times 1}$.

For the sparse GP moments, $\mathbf{X}$ is replaced with $\mathbf{Z}$ as in Equation (E.19), including inside $\boldsymbol{\psi}_1{}^*, \boldsymbol{\psi}_2{}^*$.

E.5 Additional Results

The experimental results in Section 5.6.5 are discussed, however for brevity we did not include all the relavent graphs that summarise the results of Table 5.1 and Table 5.2. We provide these informative results in this section. See the main text for the experimental setup used.

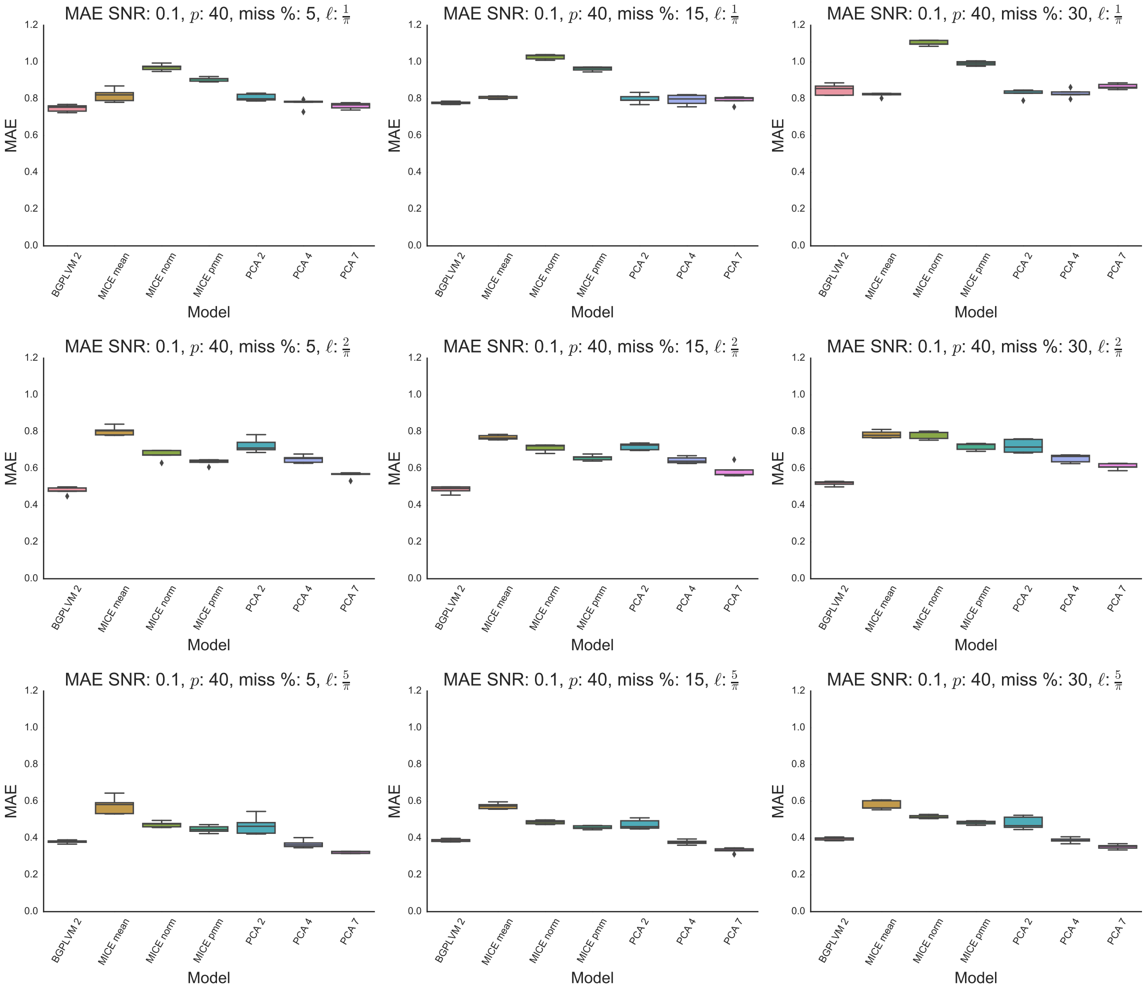


MAE SNR: 0.1, p: 40, miss %: 15, ℓ: $\frac{1}{\pi}$

MAE SNR: 0.1, p: 40, miss %: 15, ℓ: $\frac{2}{\pi}$

MAE SNR: 0.1, p: 40, miss %: 15, ℓ: $\frac{5}{\pi}$

Fig. E.1 Simulation study with Gaussian outputs, SNR = 0.01 15% missing data



MAE SNR: 0.1, p: 40, miss %: 5, ℓ: $\frac{1}{\pi}$

MAE SNR: 0.1, p: 40, miss %: 15, ℓ: $\frac{1}{\pi}$

MAE SNR: 0.1, p: 40, miss %: 30, ℓ: $\frac{1}{\pi}$

MAE SNR: 0.1, p: 40, miss %: 5, ℓ: $\frac{2}{\pi}$

MAE SNR: 0.1, p: 40, miss %: 15, ℓ: $\frac{2}{\pi}$

MAE SNR: 0.1, p: 40, miss %: 30, ℓ: $\frac{2}{\pi}$

MAE SNR: 0.1, p: 40, miss %: 5, ℓ: $\frac{5}{\pi}$

MAE SNR: 0.1, p: 40, miss %: 15, ℓ: $\frac{5}{\pi}$

MAE SNR: 0.1, p: 40, miss %: 30, ℓ: $\frac{5}{\pi}$

Fig. E.2 Simulation study with Gaussian outputs, SNR = 0.1

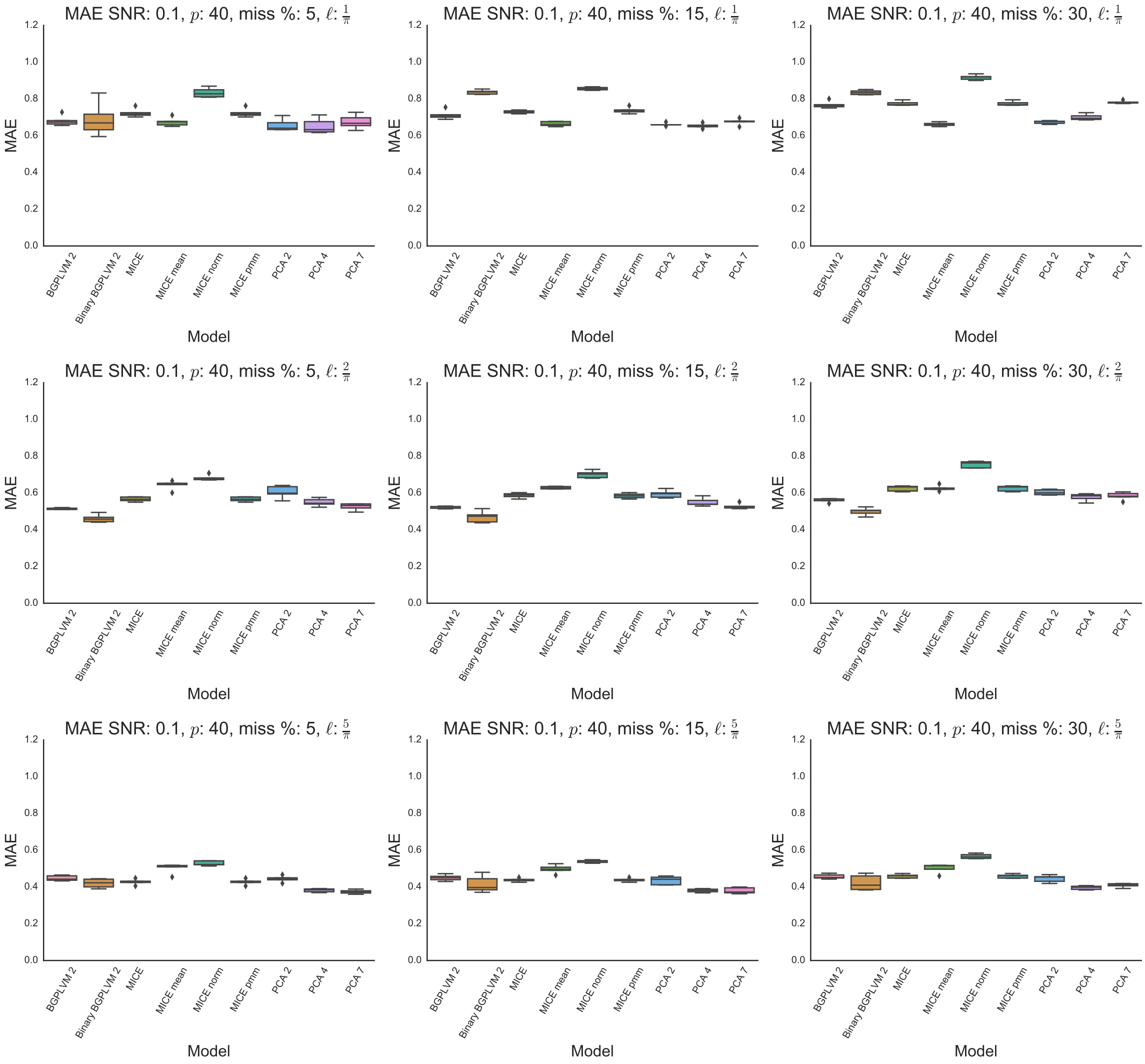


Fig. E.3 Simulation study with mixed binary and Gaussian outputs, SNR = 0.1

References

Adams, R. P. and Stegle, O. (2008). Gaussian process product models for nonparametric nonstationarity. In *25th International Conference on Machine Learning (ICML)*, pages 1–8.

Álvarez, M. A., Rosasco, L., and Lawrence, N. D. (2012). Kernels for vector-valued functions: A review. *Foundations and Trends in Machine Learning*, 4(3):195–266.

Andrade, R. (2015). *Gaussian Processes For Spatiotemporal Modelling*. PhD thesis, University of Sheffield.

Andridge, R. R. and Little, R. J. (2010). A review of hot deck imputation for survey non-response. *International Statistical Review*, 78(1):40–64.

Azur, M. J., Stuart, E. A., Frangakis, C., and Leaf, P. J. (2011). Multiple imputation by chained equations: What is it and how does it work? *International Journal of Methods in Psychiatric Research*, 20(1):40–49.

Bakker, B., Kappen, B., and Heskes, T. (2000). *Survival Analysis: A Neural-Bayesian Approach*. Springer London, London.

Barber, D. and Williams, C. K. I. (1997). Gaussian processes for Bayesian classification via hybrid Monte Carlo. In *Advances in Neural Information Processing Systems 9 (NIPS)*, pages 340–346. MIT Press.

Barlow, R. E. and Proschan, F. (1975). *Statistical theory of reliability and life testing probability models*. Holt, Rinehart and Winston Inc, New York.

Barrett, J. E. and Coolen, A. C. (2013). Gaussian process regression for survival data with competing risks. *arXiv preprint arXiv:1312.1591*.

Barrett, J. E. and Coolen, A. C. (2015). Covariate dimension reduction for survival data via the Gaussian process latent variable model. *Statistics in Medicine*, 35:1340–1353.

Biganzoli, E., Boracchi, P., Mariani, L., and Marubini, E. (1998). Feed forward neural networks for the analysis of censored survival data: a partial logistic regression approach. *Statistics in Medicine*, 17(10):1169–1186.

Bishop, C. M. (2006). *Pattern Recognition and Machine Learning*. Springer-Verlag.

Blei, D. M., Kucukelbir, A., and McAuliffe, J. D. (2016). Variational inference: A review for statisticians. *arXiv preprint arXiv:1601.00670*.

Breiman, L. (2001). Random forests. *Machine learning*, 45(1):5–32.

Breslow, N. E. (1972). Discussion of regression models and life-tables. *Journal of the Royal Statistical Society, Series B, Methodological*, 34(2):187–220.

Buettner, F., Moignard, V., Göttgens, B., and Theis, F. J. (2014). Probabilistic PCA of censored data: Accounting for uncertainties in the visualization of high-throughput single-cell qPCR data. *Bioinformatics*, 30(13):1867–1875.

Buuren, S. and Groothuis-Oudshoorn, K. (1999). *Flexible Mutlivariate Imputation By MICE*. Leiden: TNO.

Buuren, S. and Groothuis-Oudshoorn, K. (2011). Mice: Multivariate imputation by chained equations in r. *Journal of Statistical Software*, 45(3).

Candela, J. Q., Girard, A., Larsen, J., and Rasmussen, C. E. (2003). Propagation of uncertainty in Bayesian kernel models-application to multiple-step ahead forecasting. In *International Conference on Acoustics, Speech, and Signal Processing*, volume 2, pages II–701. IEEE.

Cao, Y. and Fleet, D. J. (2014). Generalized product of experts for automatic and principled fusion of Gaussian process predictions. *arXiv preprint arXiv:1410.7827*.

Cawley, G., Talbot, N., and Chapelle, O. (2006a). Estimating predictive variances with kernel ridge regression. In *MLCW 2005*, pages 56–77, Berlin, Germany. Max-Planck-Gesellschaft, Springer.

Cawley, G. C., Talbot, N. L., Foxall, R. J., Dorling, S. R., and Mandic, D. P. (2004). Heteroscedastic kernel ridge regression. *Neurocomputing*, 57:105–124.

Cawley, G. C., Talbot, N. L. C., Janacek, G. J., and Peck, M. W. (2006b). Sparse Bayesian kernel survival analysis for modelling the growth domain of microbial pathogens. *IEEE Transactions on Neural Networks*, 17(2):471–481.

Collett, D. (2015). *Modelling Survival Data In Medical Research.* CRC press.

Cox, David, R. (1972). Regression models and life-tables. *Journal of the Royal Statistical Society, Series B, Methodological*, 34(2):187–220.

Cox, David, R. (1975). Partial likelihood. *Biometrika*, 62(2):269–276.

Cox, David, R. and Oakes, D. (1984). *Analysis Of Survival Data*, volume 5. Chapman and Hall/CRC.

Csató, L. and Opper, M. (2002). Sparse on-line Gaussian processes. *Neural Computation*, 14(3):641–668.

Dai, Z., Damianou, A., Hensman, J., and Lawrence, N. D. (2014). Gaussian process models with parallelization and GPU acceleration. *arxiv preprint arXiv:1410.4984*.

Damianou, A. (2015). *Deep Gaussian Processes And Variational Propagation Of Uncertainty.* PhD thesis, University of Sheffield.

Damianou, A. and Lawrence, N. D. (2015). Semi-described and semi-supervised learning with Gaussian processes. In *31st Conference on Uncertainty in Artificial Intelligence (UAI).*

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.

Dauphin, Y. N., Pascanu, R., Gulcehre, C., Cho, K., Ganguli, S., and Bengio, Y. (2014). Identifying and attacking the saddle point problem in high-dimensional non-convex optimization. In *Advances in Neural Information Processing Systems 27 (NIPS)*, pages 2933–2941.

Deisenroth, M. P. (2010). *Efficient Reinforcement Learning Using Gaussian Processes*, volume 9. KIT Scientific Publishing.

Deisenroth, M. P. and Ng, J. W. (2015). Distributed Gaussian processes. In *32nd International Conference on Machine Learning (ICML)*, volume 2, page 5.

Dezfouli, A. and Bonilla, E. V. (2015). Scalable inference for Gaussian process models with black-box likelihoods. In *Advances in Neural Information Processing Systems 28 (NIPS)*, pages 1414–1422.

Duchi, J., Hazan, E., and Singer, Y. (2011). Adaptive subgradient methods for online learning and stochastic optimization. *Journal of Machine Learning Research*, 12:2121–2159.

Duvenaud, D. (2014). *Automatic Model Construction with Gaussian Processes*. PhD thesis, Computational And Biological Learning Laboratory, University Of Cambridge.

Efron, B. (2002). The two-way proportional hazards model. *Journal of the Royal Statistical Society: Series B, Statistical Methodology*, 64(4):899–909.

Ertin, E. (2007). Gaussian process models for censored sensor readings. In *14th Workshop on Statistical Signal Processing*, pages 665–669. IEEE.

Fan, J., Li, G., and Li, R. (2005). An overview on variable selection for survival analysis. *Contemporary multivariate analysis and design of experiments*, page 315.

Fernandez, T., Rivera, N., and Teh, Y. W. (2016). Gaussian processes for survival analysis. In *Advances in Neural Information Processing Systems 29 (NIPS)*, pages 5015–5023.

Ferrari, S. and Cribari-Neto, F. (2004). Beta regression for modelling rates and proportions. *Journal of Applied Statistics*, 31(7):799–815.

Fisher, L. D. and Lin, D. Y. (1999). Time-dependent covariates in the Cox proportional-hazards regression model. *Annual Review of Public Health*, 20(1):145–157.

Gal, Y., Chen, Y., and Ghahramani, Z. (2015). Latent Gaussian processes for distribution estimation of multivariate categorical data. In *32nd International Conference on Machine Learning (ICML)*, pages 645–654.

Gal, Y. and van der Wilk, M. (2014). Variational inference in sparse Gaussian process regression and latent variable models - a gentle tutorial. *arXiv preprint arXiv:1402.1412*.

Gal, Y., van der Wilk, M., and Rasmussen, C. E. (2014). Distributed variational inference in sparse Gaussian process regression and latent variable models. In Ghahramani, Z., Welling, M., Cortes, C., Lawrence, N. D., and Weinberger, K. Q., editors, *Advances in Neural Information Processing Systems*, volume 27, Cambridge, MA.

Gelman, A., Carlin, J. B., Stern, H. S., Dunson, D. B., Vehtari, A., and Rubin, D. B. (2013). *Bayesian Data Analysis, Third Edition*. CRC Press.

Gibbs, M. N. and MacKay, D. J. (2000). Variational Gaussian process classifiers. *IEEE Transactions On Neural Networks*, 11(6):1458–1464.

Girard, A., Rasmussen, C. E., Quinonero-Candela, J., and Murray-Smith, R. (2003). Gaussian process priors with uncertain inputs application to multiple-step ahead time series forecasting. In *Advances in Neural Information Processing Systems 15 (NIPS)*. MIT Press.

Goldberg, P. W., Williams, C. K., and Bishop, C. M. (1997). Regression with input-dependent noise: A Gaussian process treatment. In *Advances in Neural Information Processing Systems 28 (NIPS)*, volume 10, pages 493–499.

Goldberg, P. W., Williams, C. K. I., and Bishop, C. M. (1998). Regression with input-dependent noise: A Gaussian process treatment. In Jordan, M. I., Kearns, M. J., and Solla, S. A., editors, *Advances in Neural Information Processing Systems*, volume 10, pages 493–499, Cambridge, MA. MIT Press.

Goodfellow, I., Bengio, Y., and Courville, A. (2016). *Deep learning*. MIT Press.

Graham, J. W. (2009). Missing data analysis: Making it work in the real world. *Annual Review of Psychology*, 60:549–576.

Groot, P. and Lucas, P. (2012). Gaussian process regression with censored data using expectation propagation. In *6th European Workshop on Probabilistic Graphical Models*.

Hanley, J. A. and Miettinen, O. S. (2009). Fitting smooth-in-time prognostic risk functions via logistic regression. *The International Journal of Biostatistics*, 5(1).

Hastie, T., Tibshirani, R., and Friedman, J. (2009). *The Elements of Statistical Learning*. Springer-Verlag, 2nd edition.

Hastie, T. J. and Tibshirani, R. J. (1990). *Generalized Additive Models*. Chapman and Hall, London.

Henderson, R., Shimakura, S., and D., G. (2002). Modeling spatial variation in leukemia survival data. *Journal of The American Statistical Association*, 97:965–972.

Hensman, J., Fusi, N., and Lawrence, N. D. (2013). Gaussian processes for big data. In Nicholson, A. and Smyth, P., editors, *Uncertainty in Artificial Intelligence*, volume 29. AUAI Press.

Hensman, J. and Lawrence, N. D. (2014). Nested variational compression in deep Gaussian processes. *arXiv preprint arXiv:1412.1370*.

Hensman, J., Matthews, A. G., Filippone, M., and Ghahramani, Z. (2015a). MCMC for variationally sparse Gaussian processes. In *Advances in Neural Information Processing Systems 28 (NIPS)*, pages 1648–1656.

Hensman, J., Matthews, A. G. D. G., and Ghahramani, Z. (2015b). Scalable variational Gaussian process classification. In *18th International Conference on Artificial Intelligence and Statistics (AISTATS)*, pages 1–9.

Hensman, J., Zwiessele, M., and Lawrence, N. D. (2014). Tilted variational Bayes. In Kaski, S. and Corander, J., editors, *Proceedings of the Seventeenth International Workshop on Artificial Intelligence and Statistics*, volume 33, pages 356–364, Iceland. JMLR W&CP 33.

Hernández-Lobato, D. and Hernández-Lobato, J. M. (2016). Scalable Gaussian process classification via expectation propagation. In *19th International Conference on Artificial Intelligence and Statistics (AISTATS)*, pages 168–176.

Hernández-Lobato, J. M., Houlsby, N., and Ghahramani, Z. (2014). Probabilistic matrix factorization with non-random missing data. In *31st International Conference on Machine Learning (ICML)*, pages 1512–1520.

Hoerl, A. E. and Kennard, R. W. (1970). Ridge regression: Biased estimation for nonorthogonal problems. *Technometrics*, 12(1):55–67.

Hoffman, M. D., Blei, D. M., Wang, C., and Paisley, J. (2013). Stochastic variational inference. *Journal of Machine Learning Research*, 14:1303–1347.

Ibrahim, J. G., Chen, M.-H., and Sinha, D. (2005). *Bayesian Survival Analysis*. Springer, 2 edition.

Ilin, A. and Raiko, T. (2010). Practical approaches to principal component analysis in the presence of missing values. *Journal of Machine Learning Research*, 11(Jul):1957–2000.

Ishwaran, H., Kogalur, U. B., Blackstone, E. H., and Lauer, M. S. (2008). Random survival forests. *The Annals of Applied Statistics*, pages 841–860.

Jaakkola, T. S. and Jordan, M. I. (1996). Computing upper and lower bounds on likelihoods in intractable networks. In *12th International Conference on Uncertainty in Artificial Intelligence (UAI)*, pages 340–348. Morgan Kaufmann Publishers Inc.

Jiezhi, Q. (2009). *Comparison Of Proportional Hazards And Accelerated Failure Time Models*. Masters, University Of Saskatchewan.

Joensuu, H., Reichardt, P., Eriksson, M., Hall, K. S., and Vehtari, A. (2013). Gastrointestinal stromal tumor: A method for optimizing the timing of CT scans in the follow-up of cancer patients. *Radiology*, 271(1):96–106.

Joensuu, H., Vehtari, A., Riihimäki, J., Nishida, T., Steigen, S. E., Brabec, P., Plank, L., Nilsson, B., Cirilli, C., Braconi, C., Bordoni, A., K Magnusson, M., Linke, Z., Sufliarsky, J., Massimo, F., Jonasson, J. G., Paolo Dei Tos, A., and Rutkowski, P. (2012). Risk of recurrence of gastrointestinal stromal tumour after surgery: An analysis of pooled population-based cohorts. *The Lancet Oncology*, 13(3):265–274.

Jolliffe, I. (1986). *Principal Component Analysis*. Springer.

Jordan, M. I., Ghahramani, Z., Jaakkola, T. S., and Saul, L. K. (1999). An introduction to variational methods for graphical models. *Machine Learning*, 37(2):183–233.

Jylänki, P., Vanhatalo, J., and Vehtari, A. (2011). Robust Gaussian process regression with a Student-t likelihood. *Journal Of Machine Learning Research*, 12:3227–3257.

Kalbfleisch, J. D. and Prentice, R. L. (2002). *The Statistical Analysis Of Failure Time Data*. J. Wiley.

Katzman, J., Shaham, U., Cloninger, A., Bates, J., Jiang, T., and Kluger, Y. (2016). Deep survival: A deep Cox proportional hazards network. *arXiv preprint arXiv:1606.00931*.

Kersting, K., Plagemann, C., Pfaff, P., and Burgard, W. (2007). Most likely heteroscedastic Gaussian process regression. In *24th International Conference on Machine Learning (ICML)*, pages 393–400.

Khan, E., Mohamed, S., and Murphy, K. P. (2012). Fast Bayesian inference for non-conjugate Gaussian process regression. In *Advances in Neural Information Processing Systems 26 (NIPS)*, pages 3140–3148.

Khan, F. and Zubek, V. (2008). Support vector regression for censored data (svrc): A novel tool for survival analysis. In *Eighth IEEE International Conference on Data Mining (ICDM)*.

Kingma, D. and Ba, J. (2014). Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Kingma, D. P. and Welling, M. (2014). Stochastic gradient VB and the variational auto-encoder. In *2nd International Conference on Learning Representations (ICLR)*.

Klein, J. P. and Moeschberger, M. L. (2003). *Survival Analysis: Techniques For Censored And Truncated Data*. Springer Science & Business Media.

Kleinbaum, D. G. and Klein, M. (2006). *Survival Analysis: A Self-Learning Text*. Springer Science & Business Media.

Krewski, D., Burnett, R., Goldberg, M., Hoover, B. K., Siemiatycki, J., Jerrett, M., Abrahamowicz, M., and White, W. (2003). Overview of the reanalysis of the Harvard six cities study and American cancer society study of particulate air pollution and mortality. *Journal of Toxicology and Environmental Health, Part A*, 66(16-19):1507–1552.

Kullback, S. (1959). *Information Theory and Statistics*. Dover Publications, New York.

Kullback, S. and Leibler, R. A. (1951). On information and sufficiency. *Annals of Mathematical Statistics*, 22:79–86.

Kuß, M. (2006). *Gaussian Process Models For Robust Regression, Classification, And Reinforcement Learning*. PhD thesis, TU Darmstadt.

Kuss, M. and Rasmussen, C. E. (2005). Assessing approximate inference for binary Gaussian process classification. *Journal of Machine Learning Research*, 6(Oct):1679–1704.

Lawrence, N. D. (2000). *Variational Inference in Probabilistic Models*. PhD thesis, Computer Laboratory, University of Cambridge, New Museums Site, Pembroke Street, Cambridge, CB2 3QG, U.K. Available from http://www.thelawrences.net/neil.

Lawrence, N. D. (2004). Probabilistic non-linear principal component analysis with Gaussian process latent variable models. Technical Report CS-04-08, Department of Computer Science, University of Sheffield.

Lawrence, N. D. and Urtasun, R. (2009). Non-linear matrix factorization with Gaussian processes. In Bottou, L. and Littman, M., editors, *Proceedings of the International Conference in Machine Learning*, volume 26, San Francisco, CA. Morgan Kauffman.

Lazaro-Gredilla, M. and Titsias, M. (2011). Variational heteroscedastic Gaussian process regression. In *28th International Conference on Machine Learning (ICML)*, pages 841–848.

Le, Q. V., Smola, A. J., and Canu, S. (2005). Heteroscedastic Gaussian process regression. In *22nd International Conference on Machine Learning (ICML)*, pages 489–496. ACM.

Li, D., Wang, X., and Dey, D. K. (2016). A flexible cure rate model for spatially correlated survival data based on generalized extreme value distribution and Gaussian process priors. *Biometrical Journal*.

Lin, D. Y. and Ying, Z. (1995). Semiparametric analysis of general additive-multiplicative hazard models for counting processes. *The Annals of Statistics*, 23(5):1712–1734.

Linderman, S. and Adams, R. (2014). Discovering latent network structure in point process data. In *31st International Conference on Machine Learning (ICML)*.

Little, R. J. A. and Rubin, D. B. (1986). *Statistical Analysis With Missing Data*. John Wiley & Sons, Inc., New York, NY, USA.

Lloyd, C. M., Gunter, T., Osborne, M. A., and Roberts, S. J. (2015). Variational inference for Gaussian process modulated poisson processes. In Bach, F. R. and Blei, D. M., editors, *32nd International Conference on Machine Learning (ICML)*, volume 37, pages 1814–1822.

MacKay, D. J. (1996). Hyperparameters: Optimize, or integrate out? In *Maximum Entropy and Bayesian Methods*, pages 43–59. Springer.

Martino, S., Akerkar, R., and Rue, H. (2010). Approximate Bayesian inference for survival models. *Scandinavian Journal of Statistics*, 38(3):514–528.

McCullagh, P. and Nelder, J. A. (1989). *Generalized Linear models*. Chapman & Hall Ltd, London, 2nd edition.

Miller, R. G. (1976). Least squares regression with censored data. *Biometrika*, pages 449–464.

Minka, T. P. (2000). Old and new matrix algebra useful for statistics. Technical report, Microsoft Research.

Minka, T. P. (2001). *A family of algorithms for approximate Bayesian inference*. PhD thesis, Massachusetts Institute of Technology.

Møller, J., Syversveen, A. R., and Waagepetersen, R. P. (1998). Log Gaussian Cox processes. *Scandinavian Journal of Statistics*, 25(3):451–482.

Murphy, K. P. (2012). *Machine Learning: A Probabilistic Perspective*. MIT press.

Neal, R. M. (1996). *Bayesian Learning for Neural Networks*. Springer.

Neal, R. M. (1997). Monte Carlo implementation of Gaussian process models for Bayesian regression and classification. Technical report, Dept. Of Statistics, Universtiy Of Toronto.

Nelder, J. and Wedderburn, R. (1972). Generalized linear models. *Journal of the Royal Statistical Society, A*, 135(3).

Nguyen, T. V. and Bonilla, E. V. (2014). Automated variational inference for Gaussian process models. In *Advances in Neural Information Processing Systems 27 (NIPS)*, pages 1404–1412.

Nickisch, H. and Rasmussen, C. E. (2008). Approximations for binary Gaussian process classification. *Journal of Machine Learning Research*, 9(Oct):2035–2078.

Opper, M. and Archambeau, C. (2009). The variational Gaussian approximation revisited. *Neural Computation*, 21(3):786–792.

Osborne, M. (2010). *Bayesian Gaussian Processes For Sequential Prediction, Optimisation And Quadrature*. PhD thesis, University of Oxford.

Osborne, M., Rogers, A., Roberts, S., Ramchurn, S., and Jennings, N. (2010). Bayesian Gaussian process models for multi-sensor time-series prediction. *Inference and Learning in Dynamic Models*.

Patel, K., Kay, R., and Rowell, L. (2006). Comparing proportional hazards and accelerated failure time models: An application in influenza. *Pharmaceutical Statistics*, 5(3):213–224.

Piironen, J. and Vehtari, A. (2015). Projection predictive model selection for Gaussian processes. *arxiv preprint arXiv:1510.04813*.

Pocock, S. J., Travison, T. G., and Wruck, L. M. (2007). Figures in clinical trial reports: Current practice & scope for improvement. *Trials*, 8:36.

Quiñonero Candela, J. and Rasmussen, C. E. (2005). A unifying view of sparse approximate Gaussian process regression. *Journal of Machine Learning Research*, 6:1939–1959.

Quinonero-Candela, J., Rasmussen, C. E., and Williams, C. K. (2007). Approximation methods for Gaussian process regression. *Large-Scale Kernel Machines*, pages 203–224.

Ranganath, R., Gerrish, S., and Blei, D. M. (2014). Black box variational inference. In *17th International Conference on Artificial Intelligence and Statistics (AISTATS)*, pages 814–822.

Ranganath, R., Perotte, A., Elhadad, N., and Blei, D. (2016). Deep survival analysis. *arXiv preprint arXiv:1608.02158*.

Rasmussen, C. E. and Williams, C. K. I. (2006). *Gaussian Processes for Machine Learning*. MIT Press, Cambridge, MA.

Rezende, D. J., Mohamed, S., and Wierstra, D. (2014). Stochastic back-propagation and variational inference in deep latent Gaussian models. Technical report.

Riihimäki, J. and Vehtari, A. (2014). Laplace approximation for logistic Gaussian process density estimation and regression. *Bayesian Analysis*, 9(2):425–448.

Ripley, B. D. and Ripley, R. M. (2001). Neural networks as statistical methods in survival analysis. *Clinical Applications of Artificial Neural Networks*, pages 237–255.

Rosset, S., Neumann, E., Eick, U., and Vatnik, N. (2003). Customer lifetime value models for decision support. *Data Mining and Knowledge Discovery*, 7(3):321–339.

Royston, P. and Lambert, P. C. (2011). *Flexible Parametric Survival Analysis Using Stata: Beyond the Cox Model*. Stata Press.

Rubin, D. B. (2004). *Multiple Imputation For Nonresponse In Surveys*, volume 81. John Wiley & Sons.

Rue, H. and Held, L. (2005). *Gaussian Markov Random Fields: Theory And Applications*. CRC Press.

Rue, H., Martino, S., and Chopin, N. (2009). Approximate Bayesian inference for latent Gaussian models by using integrated nested Laplace approximations. *Journal of the Royal Statistical Society, Series B, Statistical Methodology*, 71(2):319–392.

Rumelhart, D. E., Hinton, G. E., and Williams, R. J. (1985). Learning internal representations by error propagation. Technical report, DTIC Document.

Saul, A. D., Hensman, J., Vehtari, A., and Lawrence, N. D. (2016). Chained Gaussian processes. In Gretton, A. and Robert, C., editors, *Proceedings of the Nineteenth International Workshop on Artificial Intelligence and Statistics*, volume 51, pages 1431–1440, Cadiz, Spain. JMLR W&CP 51.

Saunders, C., Gammerman, A., and Vovk, V. (1998). Ridge regression learning algorithm in dual variables. In *Proceedings of the Fifteenth International Conference on Machine Learning (ICML)*, volume 98, pages 515–521.

Schölkopf, B. (2000). The kernel trick for distances. In *Advances in Neural Information Processing Systems 13 (NIPS)*, pages 301–307.

Schölkopf, B. and Smola, A. (2002). *Learning With Kernels: Support Vector Machines, Regularization, Optimization, And Beyond.* Adaptive Computation And Machine Learning. MIT Press, Cambridge, MA, USA.

Seeger, M. (2000). Bayesian model selection for support vector machines, Gaussian processes and other kernel classifiers. In *Advances in Neural Information Processing Systems 13 (NIPS)*, pages 603–609.

Seeger, M., Williams, C. K. I., and Lawrence, N. D. (2003). Fast forward selection to speed up sparse Gaussian process regression. In Bishop, C. M. and Frey, B. J., editors, *Proceedings of the Ninth International Workshop on Artificial Intelligence and Statistics*, Key West, FL.

Seeger, M. W. and Nickisch, H. (2011). Fast convergent algorithms for expectation propagation approximate Bayesian inference. In *14th International Conference on Artificial Intelligence and Statistics (AISTATS)*, pages 652–660.

Sefrioui, I., Amadini, R., Mauro, J., El Fallahi, A., and Gabbrielli, M. (2017). Survival prediction of trauma patients: a study on us national trauma data bank. *European Journal of Trauma and Emergency Surgery*, pages 1–18.

Shivaswamy, P. K., Chu, W., and Jansche, M. (2007). A support vector approach to censored targets. In *Seventh IEEE International Conference on Data Mining (ICDM)*, pages 655–660.

Silverman, B. W. (1985). Some aspects of the spline smoothing approach to non-parametric regression curve fitting (with discussion). *Journal of the Royal Statistical Society, B*, 47(1):1–52.

Smith, P. J. (2002). *Analysis Of Failure And Survival Data*. CRC Press.

Snelson, E. and Ghahramani, Z. (2005). Sparse Gaussian processes using pseudo-inputs. In *Advances in Neural Information Processing Systems 18 (NIPS)*, pages 1257–1264.

Snelson, E. and Ghahramani, Z. (2006a). Sparse Gaussian processes using pseudo-inputs. In Weiss, Y., Schölkopf, B., and Platt, J. C., editors, *Advances in Neural Information Processing Systems*, volume 18, Cambridge, MA. MIT Press.

Snelson, E. and Ghahramani, Z. (2006b). Variable noise and dimensionality reduction for sparse Gaussian processes. In *22nd Conference on Uncertainty in Artificial Intelligence (UAI)*.

Snelson, E., Rasmussen, C. E., and Ghahramani, Z. (2004). Warped Gaussian processes. In Thrun, S., Saul, L., and Schölkopf, B., editors, *Advances in Neural Information Processing Systems*, volume 16, Cambridge, MA. MIT Press.

Sperrin, M. and Buchan, I. (2012). Modelling time to event with observations made at arbitrary times. *Statistics in Medicine*, 32(July 2012).

Spruance, S. L., Reid, J. E., Grace, M., and Samore, M. (2004). Hazard ratio in clinical trials. *Antimicrobial Agents and Chemotherapy*, 48(8).

Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., and Salakhutdinov, R. (2014). Dropout: A simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research*, 15:1929–1958.

Tieleman, T. and Hinton, G. (2012). RMSProp. Coursera: Neural Networks For Machine Learning.

Tipping, M. E. and Bishop, C. M. (1999). Probabilistic principal component analysis. *Journal of the Royal Statistical Society, B*, 6(3):611–622.

Titsias, M. K. (2009). Variational learning of inducing variables in sparse Gaussian processes. In van Dyk, D. and Welling, M., editors, *Proceedings of the Twelfth International Workshop on Artificial Intelligence and Statistics*, volume 5, pages 567–574, Clearwater Beach, FL. JMLR W&CP 5.

Titsias, M. K., Lawrence, N., and Rattray, M. (2008). Markov chain Monte Carlo algorithms for Gaussian processes. *Inference and Estimation in Probabilistic Time-Series Models*, 9.

Titsias, M. K. and Lawrence, N. D. (2010). Bayesian Gaussian process latent variable model. In Teh, Y. W. and Titterington, D. M., editors, *Proceedings of the Thirteenth International Workshop on Artificial Intelligence and Statistics*, volume 9, pages 844–851, Chia Laguna Resort, Sardinia, Italy. JMLR W&CP 9.

Tolvanen, V., Jylänki, P., and Vehtari, A. (2014). Expectation propagation for nonstationary heteroscedastic Gaussian process regression. In *IEEE International Workshop Machine Learning for Signal Processing (MLSP)*.

Tresp, V. (2000). A Bayesian committee machine. *Neural Computation*, 12(11):2719–2741.

Turner, R. E. and Sahani, M. (2011). Demodulation as probabilistic inference. *IEEE Transactions on Audio, Speech, and Language Processing*, 19:2398–2411.

Van Belle, V., Pelckmans, K., Van Huffel, S., and Suykens, J. A. (2011). Support vector methods for survival analysis: a comparison between ranking and regression approaches. *Artificial Intelligence in Medicine*, 53(2):107–118.

Vanhatalo, J., Pietiläinen, V., and Vehtari, A. (2010). Approximate inference for disease mapping with sparse Gaussian processes. *Statistics in Medicine*, 29(15):1580–1607.

Vanhatalo, J., Riihimäki, J., Hartikainen, J., Jylänki, P., Tolvanen, V., and Vehtari, A. (2013). GPstuff: Bayesian modeling with Gaussian processes. *Journal of Machine Learning Research*, 14(1):1175–1179. http://mloss.org/software/view/451/.

Vapnik, V. N. (1998). *Statistical Learning Theory*. Wiley-Interscience.

Vehtari, A., Tolvanen, V., Mononen, T., and Winther, O. (2014). Bayesian leave-one-out cross-validation approximations for Gaussian latent variable models. *arXiv preprint arXiv:1412.7461*.

Wan. Tang, X. T., editor (2012). *Modern Clinical Trial Analysis (Applied Bioinformatics and Biostatistics in Cancer Research)*. Springer, 2013 edition.