Variational Approximate Inference in Latent Linear Models

Edward Arthur Lester Challis

A dissertation submitted in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
of the
University of London.

Department of Computer Science
University College London

October 10, 2013

Declaration

I, Edward Arthur Lester Challis, confirm that the work presented in this thesis is my own. Where information has been derived from other sources, I confirm that this has been indicated in the thesis.

To Mum and Dad and George.

Abstract

Latent linear models are core to much of machine learning and statistics. Specific examples of this model class include Bayesian generalised linear models, Gaussian process regression models and unsupervised latent linear models such as factor analysis and principal components analysis. In general, exact inference in this model class is computationally and analytically intractable. Approximations are thus required. In this thesis we consider deterministic approximate inference methods based on minimising the Kullback-Leibler (KL) divergence between a given target density and an approximating 'variational' density.

First we consider Gaussian KL (G-KL) approximate inference methods where the approximating variational density is a multivariate Gaussian. Regarding this procedure we make a number of novel contributions: sufficient conditions for which the G-KL objective is differentiable and convex are described, constrained parameterisations of Gaussian covariance that make G-KL methods fast and scalable are presented, the G-KL lower-bound to the target density's normalisation constant is proven to dominate those provided by local variational bounding methods. We also discuss complexity and model applicability issues of G-KL and other Gaussian approximate inference methods. To numerically validate our approach we present results comparing the performance of G-KL and other deterministic Gaussian approximate inference methods across a range of latent linear model inference problems.

Second we present a new method to perform KL variational inference for a broad class of approximating variational densities. Specifically, we construct the variational density as an affine transformation of independently distributed latent random variables. The method we develop extends the known class of tractable variational approximations for which the KL divergence can be computed and optimised and enables more accurate approximations of non-Gaussian target densities to be obtained.

Acknowledgements

First I would like to thank my supervisor David Barber for the time, effort, insight and support he kindly provided me throughout this Ph.D. I would also like to thank Thomas Furmston and Chris Bracegirdle for all the helpful discussions we have had. Last but not least I would like to thank Antonia for putting up with me.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

We define the latent linear model class as consisting of those probabilistic models that describe multivariate real-valued target densities $p(\mathbf{w})$, on a vector of parameters or latent variables $\mathbf{w} \in \mathbb{R}^D$, that take the form

$$p(\mathbf{w}) = \frac{1}{Z}\mathcal{N}\left(\mathbf{w}|\boldsymbol{\mu}, \boldsymbol{\Sigma}\right) \prod_{n=1}^{N} \phi_n(\mathbf{w}^\mathsf{T}\mathbf{h}_n), \tag{1.0.1}$$

$$Z = \int \mathcal{N}\left(\mathbf{w}|\boldsymbol{\mu}, \boldsymbol{\Sigma}\right) \prod_{n=1}^{N} \phi_n(\mathbf{w}^\mathsf{T}\mathbf{h}_n) d\mathbf{w}, \tag{1.0.2}$$

where $\mathcal{N}\left(\mathbf{w}|\boldsymbol{\mu}, \boldsymbol{\Sigma}\right)$ is a multivariate Gaussian density with mean vector $\boldsymbol{\mu}$ and covariance matrix $\boldsymbol{\Sigma}$, $\mathbf{h}_n \in \mathbb{R}^D$ are fixed vectors, and $\phi_n : \mathbb{R} \to \mathbb{R}^+$ are positive, real-valued, scalar, non-Gaussian potential functions.

The latent linear model class, as defined above, is broad. In the Bayesian setting it includes Bayesian Generalised Linear Models (GLMs) such as: sparse Bayesian linear models, where the Gaussian term is the likelihood and $\{\phi_n\}_{n=1}^N$ are factors of the sparse prior [Park and Casella, 2008]; Gaussian process models, where the Gaussian term is a prior over latent function values and $\{\phi_n\}_{n=1}^N$ are factors of the non-Gaussian likelihood [Vanhatalo et al., 2009]; and binary logistic regression models, where the Gaussian term is a prior on the parameter vector and $\{\phi_n\}_{n=1}^N$ are logistic sigmoid likelihood functions [Jaakkola and Jordan, 1997]. In the context of unsupervised learning, examples include: independent components analysis, where the Gaussian term is the conditional density of the signals and $\{\phi_n\}_{n=1}^N$ are factors of the density on the latent sources $\mathbf{w}$ [Girolami, 2001]; and binary or categorical factor analysis models where the Gaussian term is the density on the latent variables and $\{\phi_n\}_{n=1}^N$ are factors of the binary or multinomial conditional distributions [Tipping, 1999, Marlin et al., 2011].

In Bayesian supervised learning, Z is the marginal likelihood, otherwise termed the evidence, and the target density $p(\mathbf{w})$ is the posterior of the parameters conditioned on the data. Evaluating Z is essential for the purposes of model comparison, hyperparameter estimation, active learning and experimental design. Indeed, any marginal function of the posterior such as a moment, or a predictive density estimate also implicitly requires Z.

In unsupervised learning, Z is the model likelihood obtained by marginalising out the hidden variables $\mathbf{w}$ and $p(\mathbf{w})$ is the density of the hidden variables conditioned on the visible variables. $p(\mathbf{w})$ is

required to optimise model parameters using either expectation maximisation or gradient ascent methods.

Computing Z, in either the Bayesian or unsupervised learning setting, is typically intractable due to the size of most problems of practical interest, which is usually much greater than one both in the dimension D and the number of potential functions N. Methods that can efficiently approximate these quantities are thus required.

Due to the importance of the latent linear model class to machine learning and statistics, a great deal of effort has been dedicated to finding accurate approximations to $p(\mathbf{w})$ and Z. Whilst there are many different possible approximation routes, including sampling, consistency methods such as expectation propagation and perturbation techniques such as the Laplace method, our focus here is on techniques that lower-bound Z and make a parametric approximation to the target density $p(\mathbf{w})$. Specifically, we obtain a parametric approximation to $p(\mathbf{w})$ and a lower-bound on $\log Z$ by minimising the Kullback-Leibler divergence between an approximating density $q(\mathbf{w})$ and the intractable density $p(\mathbf{w})$.

1.1 Contributions

Our first contributions concern Gaussian Kullback-Leibler approximate inference methods. Gaussian Kullback-Leibler (G-KL) approximate inference methods obtain a Gaussian approximation $q(\mathbf{w})$ to the target $p(\mathbf{w})$ by minimising the KL divergence $\mathrm{KL}(q(\mathbf{w})|p(\mathbf{w}))$ with respect to the moments of $q(\mathbf{w})$. Gaussian Kullback-Leibler approximate inference techniques have been known about for some time [Hinton and Van Camp, 1993, Barber and Bishop, 1998b]. However, the study and application of G-KL procedures has been limited by the perceived computational complexity of the optimisation problem they pose. We propose using a different parameterisation of the G-KL covariance than other recent treatments have considered. Doing so we are able to provide a number of novel practical and theoretical results regarding the application of G-KL procedures to latent linear models. In particular we make the following novel contributions: sufficient conditions for which the G-KL objective is differentiable and convex are described, constrained parameterisations of Gaussian covariance that make G-KL methods fast and scalable are provided, the lower-bound to the normalisation constant provided by G-KL methods is proven to dominate those provided by local variational bounding methods. For the proposed parameterisations of G-KL covariance, we discuss complexity and model applicability issues of G-KL methods compared to other Gaussian approximate inference procedures. Numerical results comparing G-KL and other deterministic Gaussian approximate inference methods are presented for: robust Gaussian process regression models with either Student's t or Laplace likelihoods, large scale Bayesian binary logistic regression models, and sequential experimental design procedures in Bayesian sparse linear models. To aide future research into latent linear models and approximate inference methods we have developed and released an open source Matlab implementation of the proposed G-KL approximate inference methods.[1]

The contributions we have made to Gaussian Kullback-Leibler approximate inference methods were presented orally at the Fourteenth International Conference on Artificial Intelligence and Statistics [Chal-

[1]The `vgai` approximate inference package is described in Appendix B.8 and can be downloaded from `mloss.org/software/view/308/`.

lis and Barber, 2011], and more recently accepted for publication in the Journal of Machine Learning Research [Challis and Barber, 2013].

Our second major contribution is a novel method to perform Kullback-Leibler approximate inference for a broad class of approximating densities $q(\mathbf{w})$. In particular, for latent linear model target densities we describe approximating densities formed from an affine transformation of independently distributed latent random variables. We refer to this set of approximating distributions as the affine independent density class. The methods we present significantly increase the set of approximating distributions for which KL approximate inference methods can be performed. Since these methods allow us to optimise the KL objective over a broader class of approximating densities they can provide more accurate inferences than previous techniques.

Our contributions concerning the affine independent KL approximate inference procedure were published in the proceedings of the Twenty Fifth Conference on Advances in Neural Information Processing Systems [Challis and Barber, 2012].

1.2 Structure of thesis

In Chapter 2 we present an introduction and overview of latent linear models. First, in Section 2.1, we consider two simple prototypical examples of latent linear models for which exact inference is analytically tractable. We then consider, in Section 2.2, various extensions to these models and the need for approximate inference methods. In light of this, in Section 2.3 we define the general form of the inference problem this thesis focusses on solving.

In Chapter 3 we provide an overview of the most commonly used deterministic approximate inference methods for latent linear models. Specifically we consider the MAP approximation, the Laplace approximation, the mean field bounding method, the Gaussian Kullback-Leibler bounding method, the local variational bounding method, and the expectation propagation approximation. For each method we consider its accuracy, speed and scalability and the range of models to which it can be applied.

In Chapter 4 we present our contributions regarding Gaussian Kullback-Leibler approximate inference routines in latent linear models. In Section 4.2 we consider the G-KL bound optimisation problem providing conditions for which the G-KL bound is differentiable and concave. In Section 4.3 we then go on to consider the complexity of the G-KL procedure, presenting efficient constrained parameterisations of covariance that make G-KL procedures fast and scalable. In Section 4.3 we compare G-KL approximate inference to other deterministic approximate inference methods, showing that the G-KL lower-bound to Z dominates the local variational lower-bound. We also discuss the complexity and model applicability issues of G-KL methods versus other Gaussian approximate inference routines.

In Chapter 5 we numerically validate the theoretical results presented in the previous chapter by comparing G-KL and other deterministic Gaussian approximate inference methods to a selection of probabilistic models. Specifically we perform experiments in robust Gaussian process regression models with either Student's t or Laplace likelihoods, large scale Bayesian binary logistic regression models, and Bayesian sparse linear models for sequential experimental design. The results confirm that G-KL methods are highly competitive versus other Gaussian approximate inference methods with regard to

both accuracy and computational efficiency.

In Chapter 6 we present a novel method to optimise the KL bound for latent linear model target densities over the class of affine independent variational densities. In Section 6.2 we introduce and describe the affine independent distribution class. In Section 6.3 we present a numerical method to efficiently evaluate and optimise the KL bound for AI variational densities. In Section 6.7 we present results showing the benefits of this procedure.

In Chapter 7 we summarise our core findings and discuss how these contributions fit within the broader context of the literature.

Chapter 2

Latent linear models

In this chapter we provide an introduction and overview of the latent linear model class. First, in Section 2.1, we consider two latent linear models for which exact inference is analytically tractable: a supervised Bayesian linear regression model and an unsupervised latent factor analysis model. These two simple models serve as archetypes by which we can introduce and discuss the core inferential quantities that this thesis is concerned with evaluating. We then consider, in Section 2.2, various generalisations of these models that render exact inference analytically intractable. In light of this, in Section 2.3, we present a specific functional form for the inference problems that we seek to address, and describe and motivate the core characteristics and trade offs by which we will measure the performance of an approximate inference method.

2.1 Latent linear models : exact inference

Latent linear models, as defined in this thesis, typically refer to either a Bayesian supervised learning model or an unsupervised latent variable model. In this section we introduce one example from each of these model classes for which exact inference is analytically tractable: a Bayesian linear regression model and an unsupervised factor analysis model.

2.1.1 Bayesian linear regression

Linear regression is one of the most popular data modelling techniques in machine learning and statistics. Linear regression assumes a linear functional relation between a vector of covariates, $\mathbf{x} \in \mathbb{R}^D$, and the mean of a scalar dependent variable $y \in \mathbb{R}$. Equivalently, linear regression assumes

$$y = \mathbf{w}^\mathsf{T}\mathbf{x} + \epsilon,$$

where $\mathbf{w} \in \mathbb{R}^D$ is the vector of parameters, and ϵ is independent additive noise with zero mean and fixed constant variance. In this section, we make the additional and common assumption that ϵ is Gaussian distributed, so that $\epsilon \sim \mathcal{N}\left(0, s^2\right)$.

The linear regression model is linear with respect to the parameters $\mathbf{w}$. The linear model can be used to represent a non-linear relation between the covariates $\mathbf{x}$ and the dependent variable y by transforming the covariates using non-linear basis functions. Transforming $\mathbf{x} \to \tilde{\mathbf{x}}$ such that $\tilde{\mathbf{x}}^\mathsf{T} := [b_1(\mathbf{x}), ..., b_K(\mathbf{x})]^\mathsf{T}$ where each $b_k : \mathbb{R}^D \to \mathbb{R}$ is a non-linear basis function, the linear model $y = \tilde{\mathbf{w}}^\mathsf{T}\tilde{\mathbf{x}} + \epsilon$ can then describe a

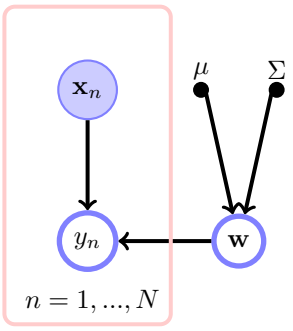


Figure 2.1: Graphical model representation of the Bayesian linear regression model. The shaded node $\mathbf{x}_n$ denotes the n^{th} observed covariate vector, and y_n the corresponding dependent variable. The plate denotes the factorisation over the N i.i.d. points in the dataset. The parameter vector $\mathbf{w}$ is an unobserved Gaussian random variable with prior $\mathbf{w} \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ and (deterministic) hyperparameters $\boldsymbol{\mu}, \boldsymbol{\Sigma}$.

non-linear relation between y and $\mathbf{x}$ whilst remaining linear in the parameters $\tilde{\mathbf{w}} \in \mathbb{R}^K$. In what follows we ignore any distinction between $\mathbf{x}$ and $\tilde{\mathbf{x}}$, assuming that any necessary non-linear transformations have been applied, and denote the transformed or untransformed covariates simply as $\mathbf{x}$.

Likelihood

Under the assumptions described above, and assuming that the data points, $\mathcal{D} = \{(\mathbf{x}_n, y_n)\}_{n=1}^N$, are independent and identically distributed (i.i.d.) given the parameters, the likelihood of the data is defined by the product

$$p(\mathbf{y}|\mathbf{X}, \mathbf{w}, s) = \prod_{n=1}^{N} \mathcal{N}\left(y_n|\mathbf{w}^\mathsf{T}\mathbf{x}_n, s^2\right),$$

where $\mathbf{y} := [y_1, ..., y_N]^\mathsf{T}$ and $\mathbf{X} := [\mathbf{x}_1, ..., \mathbf{x}_N]$. Note that the likelihood is a density over only the dependent variables y_n. This reflects the assumptions of the linear regression model which seeks to capture only the conditional relation between $\mathbf{x}$ and y.

Maximum likelihood estimation

The Maximum Likelihood (ML) parameter estimate, $\mathbf{w}_{ML}$, can be found by solving the optimisation problem

$$\begin{aligned}\mathbf{w}_{ML} &:= \operatorname*{argmax}_{\mathbf{w}} p(\mathbf{y}|\mathbf{X}, \mathbf{w}, s) \\ &= \operatorname*{argmax}_{\mathbf{w}} \sum_{n=1}^{N} \log \mathcal{N}\left(y_n|\mathbf{w}^\mathsf{T}\mathbf{x}_n, s^2\right) = \operatorname*{argmin}_{\mathbf{w}} \sum_{n=1}^{N} \left(y_n - \mathbf{w}^\mathsf{T}\mathbf{x}_n\right)^2. \end{aligned} \tag{2.1.1}$$

The first equality in equation (2.1.1) is due to $\log x$ being a monotonically increasing function in x. The second equality can be obtained by dropping the additive constants and the multiplicative scaling terms that are invariant to the optimisation problem. Equation (2.1.1) shows, under the additive Gaussian noise assumption, that the ML estimate coincides with the least squares solution. Differentiating the least squares objective *w.r.t.* $\mathbf{w}$ and equating the derivative to zero we obtain the standard normal equations

$$\left(\sum_{n=1}^{N} \mathbf{x}_n \mathbf{x}_n^\mathsf{T}\right) \mathbf{w}_{ML} = \left(\sum_{n=1}^{N} y_n \mathbf{x}_n\right) \quad \Leftrightarrow \quad \mathbf{w}_{ML} = \left(\mathbf{X}\mathbf{X}^\mathsf{T}\right)^{-1} \mathbf{X}\mathbf{y}.$$

Uniqueness for $\mathbf{w}_{ML}$ requires that $\mathbf{X}\mathbf{X}^\mathsf{T}$ is invertible, that is we require that $N \geq D$ and the data points span $\mathbb{R}^D$. Even when these conditions are satisfied, however, if $\mathbf{X}\mathbf{X}^\mathsf{T}$ is poorly conditioned the maximum likelihood solution can be unstable. We say that a matrix is poorly conditioned if its condition number is

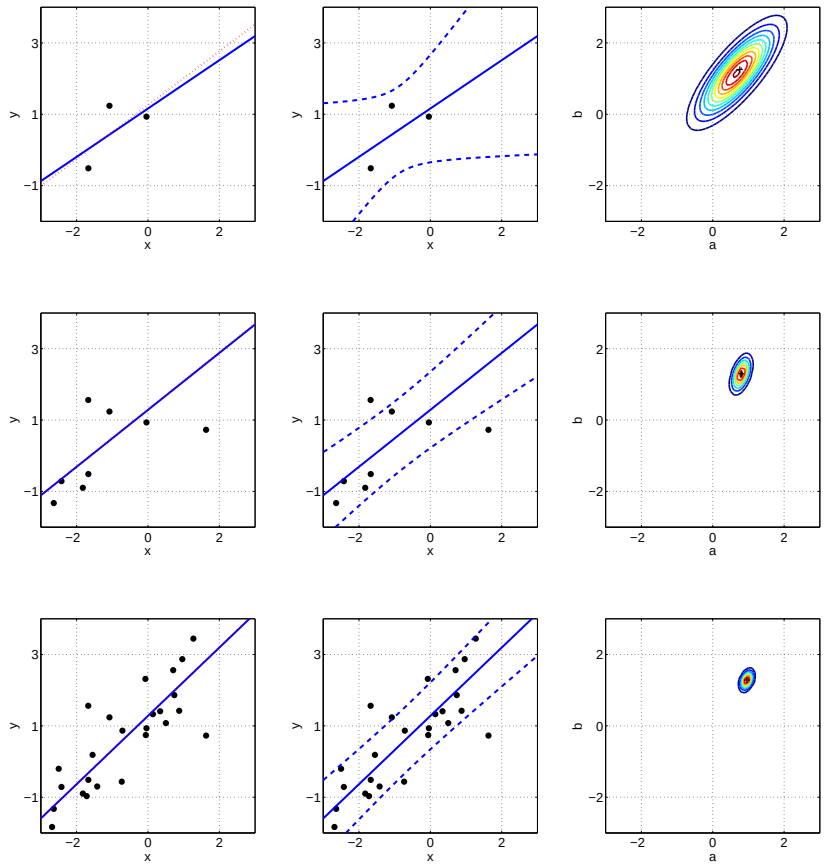


Figure 2.2: Linear regression in the model $y = ax + b + \epsilon$, with dependent variables y, covariates x, regression parameters a, b and additive Gaussian noise ϵ. The dataset size, N, in the first, second and third rows is $3, 9$ and 27 respectively. The data covariates, x, are sampled from $U[-2.5, 2.5]$ and the data generating parameters are $a = b = 1$. The training points y are sampled $y \sim \mathcal{N}(a + bx, 0.6)$. In Column 1 we plot the data points (black dots), the Bayesian mean (blue solid line) and the maximum likelihood (red dotted line) predicted estimates of y. In Column 2 we plot the Bayesian mean with ± 1 standard deviation error bars of the predicted values for y. In Column 3 we plot contours of the posterior density on a, b with the maximum likelihood parameter estimate located at the black $+$ marker. As the size of the training set, N, increases the location of the posterior's mode and the maximum likelihood estimate converge and the posterior's variance decreases.

high, where the condition number of a matrix is defined as the ratio of its largest and smallest eigenvalues. When $\mathbf{X}\mathbf{X}^{\mathsf{T}}$ is poorly conditioned the ML solution can be numerically unstable, due to rounding errors, and statistically unstable, since small perturbations of the data can result in large changes in $\mathbf{w}_{ML}$. As we see below, the Bayesian approach to linear regression can alleviate these stability issues, provide error bars on predictions and can help perform tasks such as model selection, hyperparameter estimation and active learning.

Bayesian linear regression

In a Bayesian approach to the linear regression model we treat the parameters $\mathbf{w}$ as random variables and specify a prior density on them. The prior should encode any knowledge we have about the range and relative likelihood of the values that $\mathbf{w}$ can assume before having seen the data.

A commonly used and analytically convenient prior for $\mathbf{w}$ in the linear regression model considered above is a multivariate Gaussian. Due to the closure properties of Gaussian densities, for a Gaussian prior on $\mathbf{w}$, such that $p(\mathbf{w}) = \mathcal{N}(\mathbf{w}|\boldsymbol{\mu}, \boldsymbol{\Sigma})$, the joint density of the parameters $\mathbf{w}$ and dependent variables $\mathbf{y}$ is Gaussian distributed also. In this sense the Gaussian prior is conjugate for the Gaussian likelihood model. The joint density of the random variables is then defined as

$$p(\mathbf{w}, \mathbf{y}|\mathbf{X}, \boldsymbol{\mu}, \boldsymbol{\Sigma}, s) = \mathcal{N}\left(\mathbf{y}|\mathbf{X}^\mathsf{T}\mathbf{w}, s^2\mathbf{I}_N\right)\mathcal{N}(\mathbf{w}|\boldsymbol{\mu}, \boldsymbol{\Sigma}), \tag{2.1.2}$$

where $\mathbf{I}_N$ denotes the N-dimensional identity matrix. From this joint specification of the random variables in the model we may perform standard Gaussian inference operations, see Appendix A.2, to compute probabilities of interest: the marginal likelihood of the model $p(\mathbf{y}|\mathbf{X}, \boldsymbol{\mu}, \boldsymbol{\Sigma}, s)$, obtained from marginalising out the parameters $\mathbf{w}$; the posterior of the parameters conditioned on the observed data $p(\mathbf{w}|\mathbf{y}, \mathbf{X}, \boldsymbol{\mu}, \boldsymbol{\Sigma}, s)$, obtained from conditioning on $\mathbf{y}$; and the predictive density $p(y_*|\mathbf{x}_*, \mathbf{X}, \mathbf{y}, \boldsymbol{\mu}, \boldsymbol{\Sigma}, s)$ given a new covariate vector $\mathbf{x}_*$, obtained from marginalising out the parameters from the product of the posterior and the likelihood. In the following subsections we consider each of these quantities in turn, discussing both how they are used and how they are computed.

Marginal likelihood

The marginal likelihood is obtained by marginalising out the parameters $\mathbf{w}$ from the joint density defined in equation (2.1.2). Since the joint density is multivariate Gaussian the marginal likelihood is a Gaussian evaluated at $\mathbf{y}$

$$p(\mathbf{y}|\mathbf{X}, \boldsymbol{\mu}, \boldsymbol{\Sigma}, s) = \mathcal{N}\left(\mathbf{y}|\mathbf{X}^\mathsf{T}\boldsymbol{\mu}, \mathbf{X}^\mathsf{T}\boldsymbol{\Sigma}\mathbf{X} + s^2\mathbf{I}_N\right). \tag{2.1.3}$$

Taking the logarithm of equation (2.1.3) we obtain the log marginal likelihood which can be written

$$\begin{aligned}\log p(\mathbf{y}|\mathbf{X}, \boldsymbol{\mu}, \boldsymbol{\Sigma}, s) = -\frac{1}{2}\Big[&N\log(2\pi) + \log\det\left(\mathbf{X}^\mathsf{T}\boldsymbol{\Sigma}\mathbf{X} + s^2\mathbf{I}_N\right) \\ &+ \left(\mathbf{y} - \mathbf{X}^\mathsf{T}\boldsymbol{\mu}\right)^\mathsf{T}\left(\mathbf{X}^\mathsf{T}\boldsymbol{\Sigma}\mathbf{X} + s^2\mathbf{I}\right)^{-1}\left(\mathbf{y} - \mathbf{X}^\mathsf{T}\boldsymbol{\mu}\right)\Big].\end{aligned} \tag{2.1.4}$$

Directly evaluating the expression above requires us to solve a symmetric $N \times N$ linear system and compute the determinant of an $N \times N$ matrix; both computations scale $O\left(N^3\right)$ which may be infeasible when $N \gg 1$. An alternative, and possibly cheaper to evaluate, form for the marginal likelihood can be derived by collecting first and second order terms of $\mathbf{w}$ in the exponent of equation (2.1.2), completing the square and integrating – a procedure we describe in Appendix A.2. Carrying this out and taking the logarithm of the result, we obtain the following alternative form for the log marginal likelihood

$$\begin{aligned}\log p(\mathbf{y}|\mathbf{X}, \boldsymbol{\mu}, \boldsymbol{\Sigma}, s) = -\frac{1}{2}\Big[&\log\det(2\pi\boldsymbol{\Sigma}) + N\log(2\pi s^2) \\ &+\boldsymbol{\mu}^\mathsf{T}\boldsymbol{\Sigma}^{-1}\boldsymbol{\mu} + \frac{1}{s^2}\mathbf{y}^\mathsf{T}\mathbf{y} - \mathbf{m}^\mathsf{T}\mathbf{S}^{-1}\mathbf{m} - \log\det(2\pi\mathbf{S})\Big],\end{aligned} \tag{2.1.5}$$

where the vector $\mathbf{m}$ and the symmetric positive definite matrix $\mathbf{S}$ are given by

$$\mathbf{S} = \left[\boldsymbol{\Sigma}^{-1} + \frac{1}{s^2}\mathbf{X}\mathbf{X}^\mathsf{T}\right]^{-1}, \quad \text{and} \quad \mathbf{m} = \mathbf{S}\left[\frac{1}{s^2}\mathbf{X}\mathbf{y} + \boldsymbol{\Sigma}^{-1}\boldsymbol{\mu}\right]. \tag{2.1.6}$$

Computing the determinant and the inverse of general unstructured matrices scales cubically with respect to the dimension of the matrix. However, since the matrix determinant and matrix inverse terms in equation (2.1.5) and equation (2.1.4) have a special structure either form can be computed in $O\left(ND\min\left\{N, D\right\}\right)$ time by making use of the matrix inversion lemma. To see this we focus on just computing the second form, equation (2.1.5), since the matrix $\mathbf{S}$, as defined in equation (2.1.6), is also required to define the posterior density on the parameters $\mathbf{w}$.

The computational bottleneck when evaluating the marginal likelihood in equation (2.1.5) is the evaluation of $\mathbf{S}$ and $\log\det\left(\mathbf{S}\right)$ with $\mathbf{S}$ as defined in equation (2.1.6). Provided the covariance $\boldsymbol{\Sigma}$ has some structure that can be exploited so that its inverse can be computed efficiently, for example it is diagonal or banded, then these terms (and so also the marginal likelihood) can be computed in $O\left(DN\min\left\{D, N\right\}\right)$ time. For example, if $D < N$ we should first compute $\mathbf{S}^{-1}$ using equation (2.1.6) which will scale $O\left(ND^2\right)$ and then we can compute $\mathbf{S}$ and $\log\det\left(\mathbf{S}\right)$ which will scale $O\left(D^3\right)$. Alternatively, when $D > N$ we can apply the matrix inversion lemma to equation (2.1.6) to obtain

$$\mathbf{S} = \boldsymbol{\Sigma} - \boldsymbol{\Sigma}\mathbf{X}\left(s^2\mathbf{I}_N + \mathbf{X}\boldsymbol{\Sigma}\mathbf{X}^\mathsf{T}\right)^{-1}\mathbf{X}^\mathsf{T}\boldsymbol{\Sigma},$$

whose computation scales $O\left(DN^2\right)$. Similarly, the matrix determinant lemma, an identity that can derived from the matrix inversion lemma, can be used to evaluate $\log\det\left(\mathbf{S}\right)$ in $O\left(DN^2\right)$ time – see Appendix A.6.3 for the general form of the matrix inversion and determinant lemmas.

The marginal likelihood is the probability density of the dependent variables $\mathbf{y}$ conditioned on our modelling assumptions and the covariates $\mathbf{X}$. Other names for this quantity include the evidence, the partition function, or the posterior normalisation constant. The marginal likelihood can be used as a yardstick by which to asses the validity of our modelling assumptions upon having observed the data and so can be used as a means to perform model selection. If we assume two models, $\mathcal{M}_1$ and $\mathcal{M}_2$, are a priori equally likely, $p(\mathcal{M}_1) = p(\mathcal{M}_2)$, and that the models are independent of the covariates, $p(\mathcal{M}_i|\mathbf{X}) = p(\mathcal{M}_i)$, then the ratio of the model posterior probabilities is equal to the ratio of their marginal likelihoods: $p(\mathcal{M}_1|\mathbf{y}, \mathbf{X})/p(\mathcal{M}_2|\mathbf{y}, \mathbf{X}) = p(\mathbf{y}|\mathbf{X}, \mathcal{M}_1)/p(\mathbf{y}|\mathbf{X}, \mathcal{M}_2)$. In this manner we can use the marginal likelihood to make comparative statements about which of a selection of models is more likely to have generated the data and so perform model selection.

Beyond performing discrete model selection, the marginal likelihood can also be used to select between a continuum of models defined by a continuous 'hyperparameter'. A proper Bayesian treatment for any unknown parameters should be one of specifying a prior and performing inference through marginalisation and conditioning. However, specifying priors on hyperparameters often becomes impractical since the integrals that are required to perform inferences are intractable. For example consider the case where we place a prior on the variance of the additive Gaussian noise such that $s \sim p(s)$, then the marginal likelihood of the data would be defined by the integral

$$p(\mathbf{y}|\mathbf{X}, \boldsymbol{\Sigma}, \boldsymbol{\mu}) = \int\int \mathcal{N}\left(\mathbf{y}|\mathbf{X}^\mathsf{T}\mathbf{w}, s^2\mathbf{I}_N\right)\mathcal{N}\left(\mathbf{w}|\boldsymbol{\mu}, \boldsymbol{\Sigma}\right)p(s)d\mathbf{w}ds,$$

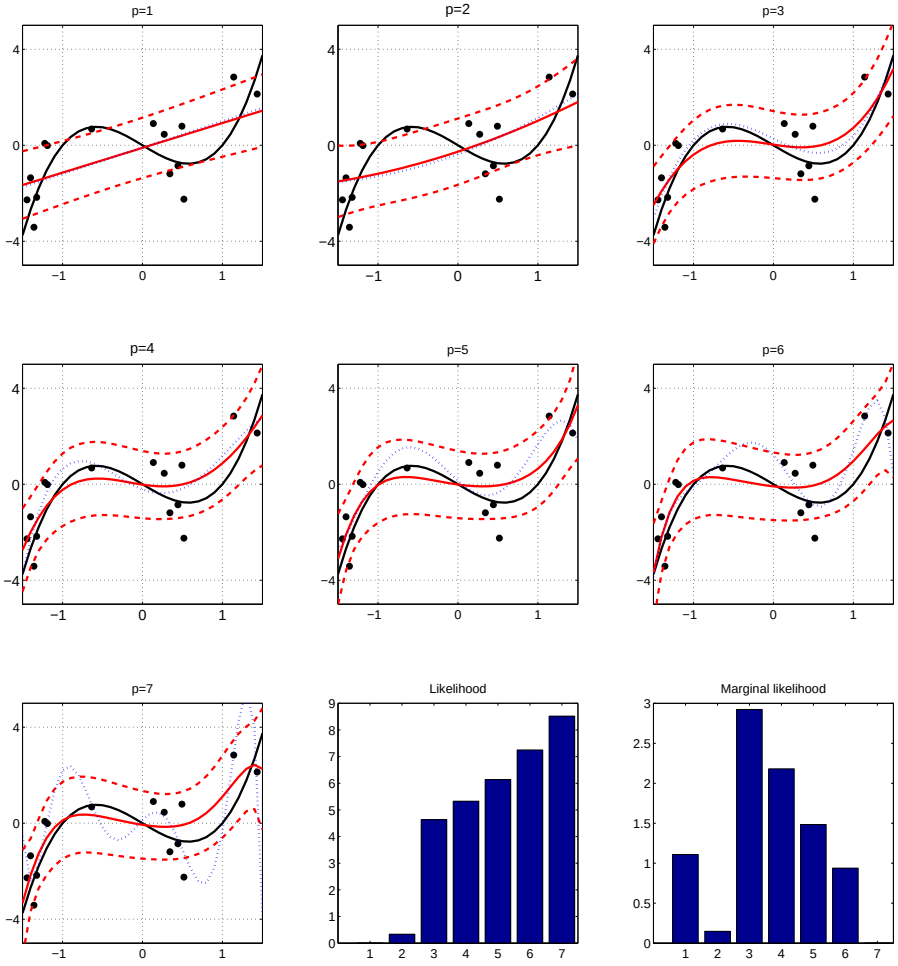


Figure 2.3: Bayesian model selection in the polynomial regression model $y = \sum_{d=0}^{P} w_d x^d + \epsilon$, with dependent variables y, covariates x, regression parameters $[w_0, ..w_P]$, and additive Gaussian noise $\epsilon \sim \mathcal{N}(0, 1)$. Standard normal factorising Gaussian priors are placed on parameters: $w_d \sim \mathcal{N}(0, 1)$. The data generating function is $y = 2x(x-1)(x+1)$. In figures $1-7$ the data (black dots), data generating function (solid black line), maximum likelihood prediction (blue dotted line), and Bayesian predicted mean (red solid line) with ± 1 standard deviation error bars (red dashed line) are plotted as we increase the order P of the polynomial regression. The likelihood increases monotonically as the model order P increases. The marginal likelihood is maximal for the true underlying model order. Likelihood and marginal likelihood values are normalised by subtracting the smallest value obtained across the models.

which for general densities on s will be intractable. However, we might expect that since the parameter s is shared by all the data points and its dimension is small compared to the data its posterior $p(s|\mathbf{y}, \mathbf{X}, \boldsymbol{\Sigma}, \boldsymbol{\mu})$ density may be reasonably approximated by a delta function centred at the mode of the likelihood $p(\mathbf{y}|\mathbf{X}, \boldsymbol{\Sigma}, \boldsymbol{\mu}, s)$.

This procedure, of performing maximum likelihood estimation on hyperparameters, is referred to as empirical Bayes or Maximum Likelihood II (ML-II). ML-II procedures are typically implemented

by numerically optimising the marginal likelihood with respect to the hyperparameters [MacKay, 1995, Berger, 1985]. In the example considered here we could use the empirical Bayes procedure to optimise the marginal likelihood over the hyperparameters $\boldsymbol{\mu}, \boldsymbol{\Sigma}$ and s which define the model's likelihood and prior densities.

The marginal likelihood naturally encodes a preference for simpler explanations of the data. This is commonly referred to as the Occam's razor effect of Bayesian model selection. Occam's principle being that given multiple hypotheses that could explain a phenomenon one should prefer that which requires the fewest assumptions. If two models, a complex one and a simple one, have similar likelihoods when applied to the same data the marginal likelihood will generally be greater for the simpler model. See MacKay [1992] for an intuitive explanation of the marginal likelihood criterion and the Occam's razor effect for model selection in linear regression models. In Figure 2.3 we show this phenomenon at work in a toy polynomial regression problem.

Posterior density

From Bayes' rule the density of the parameters $\mathbf{w}$ conditioned on the observed data is given by,

$$p(\mathbf{w}|\mathbf{y}, \mathbf{X}, \boldsymbol{\mu}, \boldsymbol{\Sigma}, s) = \frac{\mathcal{N}(\mathbf{w}|\boldsymbol{\mu}, \boldsymbol{\Sigma})\mathcal{N}\left(\mathbf{y}|\mathbf{X}^{\mathsf{T}}\mathbf{w}, s^2\mathbf{I}_N\right)}{p(\mathbf{y}|\mathbf{X}, \boldsymbol{\mu}, \boldsymbol{\Sigma}, s)} = \mathcal{N}(\mathbf{w}|\mathbf{m}, \mathbf{S}), \tag{2.1.7}$$

where the moments of the Gaussian posterior, $\mathbf{m}$ and $\mathbf{S}$, are defined in equation (2.1.6).

To gain some intuition about the posterior density in equation (2.1.7) we now consider the special case where the prior has zero mean and isotropic covariance so that $\boldsymbol{\Sigma} = \sigma^2\mathbf{I}$. For this restricted form the Gaussian posterior has mean $\mathbf{m}$ and covariance $\mathbf{S}$ given by

$$\mathbf{S} = \left[\frac{1}{\sigma^2}\mathbf{I}_D + \frac{1}{s^2}\mathbf{X}\mathbf{X}^{\mathsf{T}}\right]^{-1} \quad \text{and} \quad \mathbf{m} = \frac{1}{s^2}\mathbf{S}\mathbf{X}\mathbf{y}.$$

Inspecting these moments, we see that the mean $\mathbf{m}$ is similar to the maximum likelihood estimate. When the prior precision, $\frac{1}{\sigma^2}$, tends to zero (corresponding to an increasingly uninformative or flat prior) the posterior mean will converge to the maximum likelihood solution. Similarly, as the number of data points increases the posterior will converge to a delta function centred at the maximum likelihood solution. However, when there is limited data relative to the dimensionality of the parameter space, the prior acts as a regulariser biasing parameter estimates towards the prior's mean. The presence of the identity matrix term in $\mathbf{S}$ ensures that the posterior is stable and well defined even when $N \ll D$. See Figure 2.2 for a comparison of Bayesian and maximum likelihood parameter estimates in a toy two parameter linear regression model.

The posterior moments $\mathbf{m}, \mathbf{S}$ represent all the information the model has in the parameters $\mathbf{w}$ conditioned on the data. The vector $\mathbf{m}$ is the mean, median and mode of the posterior density since these points coincide for multivariate Gaussians. It encodes a point representation of the posterior density. The posterior covariance matrix, $\mathbf{S}$, encodes how uncertain the model is about $\mathbf{w}$ as we move away from $\mathbf{m}$. More concretely, ellipsoids in parameter space, defined by $(\mathbf{w} - \mathbf{m})^{\mathsf{T}}\mathbf{S}^{-1}(\mathbf{w} - \mathbf{m}) = c$, will have equiprobable likelihoods. For example if $\mathbf{x}$ is a unit eigenvector of the posterior covariance such that $\mathbf{S}\mathbf{x} = \lambda\mathbf{x}$ then $\mathrm{var}(\mathbf{w}^{\mathsf{T}}\mathbf{x}) = \lambda$. Analysing the posterior covariance in this fashion, we can select

directions in parameter space in which the model is least certain. Thus the posterior covariance can be used to drive sequential experimental design and active learning procedures – see for example Seo et al. [2000], Chaloner and Verdinelli [1995]. In Section 5.4 we present results from an experiment where the (approximate) Gaussian posterior covariance matrix is used to drive sequential experimental design procedures in sparse latent linear models.

Predictive density estimate

We also require the posterior density to evaluate the predictive density of an unobserved dependent variable y_* given a new covariate vector $\mathbf{x}_*$. From the conditional independence structure of the linear regression model, see Figure 2.1.2, we see that y_* is conditionally independent of the other data points $\mathbf{X}, \mathbf{y}$ given the parameters $\mathbf{w}$. Thus the predictive density estimate is defined as

$$\begin{aligned} p(y_*|\mathbf{x}_*, \mathbf{X}, \mathbf{y}) &= \int p(y_*|\mathbf{x}_*, \mathbf{w}) p(\mathbf{w}|\mathbf{X}, \mathbf{y}) d\mathbf{w} \\ &= \int \mathcal{N}\left(y_*|\mathbf{w}^\mathsf{T}\mathbf{x}_*, s^2\right) \mathcal{N}\left(\mathbf{w}|\mathbf{m}, \mathbf{S}\right) d\mathbf{w} \\ &= \mathcal{N}\left(y_*|\mathbf{m}^\mathsf{T}\mathbf{x}_*, \mathbf{x}_*^\mathsf{T}\mathbf{S}\mathbf{x}_* + s^2\right), \end{aligned}$$

where we have omitted conditional dependencies on the hyperparameters $s, \boldsymbol{\mu}, \boldsymbol{\Sigma}$ for a cleaner notation. The mean of the prediction for y_* is $\mathbf{m}^\mathsf{T}\mathbf{x}_*$ and so will converge to the maximum likelihood predicted estimate of y in the limit of large data. However, unlike in the maximum likelihood treatment, the Bayesian approach to linear regression models our uncertainty in y_* as represented by the predictive variance $\mathrm{var}(y_*) = \mathbf{x}_*^\mathsf{T}\mathbf{S}\mathbf{x}_* + s^2$. Quantifying uncertainty in our predictions is useful if we wish to minimise some asymmetric predictive loss score – for instance if over-estimation is penalised more severely than under-estimation.

Bayesian utility estimation

Inferring the posterior in a Bayesian model is typically an intermediate operation required so that we can make a decision in light of the observed data. Mathematically, for a loss $L(\mathbf{a}, \mathbf{w})$ that returns the cost of taking action $\mathbf{a} \in \mathcal{A}$ when the true unknown parameter is $\mathbf{w}$, the optimal Bayesian action, $\mathbf{a}^*$, is defined as

$$\mathbf{a}^* = \underset{\mathbf{a}\in\mathcal{A}}{\mathrm{argmin}} \int p(\mathbf{w}|\mathcal{D}, \mathcal{M}) L(\mathbf{a}, \mathbf{w}) d\mathbf{w}, \tag{2.1.8}$$

where $p(\mathbf{w}|\mathcal{D}, \mathcal{M})$ is the posterior of the parameter $\mathbf{w}$ conditioned on the data $\mathcal{D}$ and the model assumptions $\mathcal{M}$. For the Bayesian linear regression model considered here the posterior is as defined in equation (2.1.7). For example, in the forecasting setting the action $\mathbf{a}$ is the prediction $\hat{y}$ that we wish to make and the loss function returns the cost associated with over and under prediction of y.

If the action space, $\mathcal{A}$, in equation (2.1.8) is equivalent to parameter space, $\mathcal{A} = \mathcal{W}$, and if the loss function is the squared error, $L(\mathbf{a}, \mathbf{w}) := \|\mathbf{a} - \mathbf{w}\|^2$, then the optimal Bayesian parameter estimate is the posterior's mean $\mathbf{a}^* = \mathbf{m}$. For the $0-1$ loss function, $L(\mathbf{a}, \mathbf{w}) := \delta(\mathbf{a} - \mathbf{w})$ where $\delta(x)$ is the Dirac delta function, the optimal Bayes parameter estimate is the posterior's mode. To render practical the Bayesian utility approach to making decisions, we require that the integral in equation (2.1.8) is

tractable. For the Bayesian linear regression model we consider here, the posterior is Gaussian and so such expectations can often be efficiently computed – in Appendix A.2 we provide analytic expressions for a range of Gaussian expectations.

Summary

As we have seen above, Gaussian conjugacy in the Bayesian linear regression model results in compact analytic forms for many inferential quantities of interest. The joint density of all random variables in this model is multivariate Gaussian and so marginals, conditionals and first and second order moments are all immediately accessible. Specifically, closed form analytic expressions for the marginal likelihood and posterior density of the parameters conditioned on the data exist and can be computed in $O\left(ND\min\left\{D,N\right\}\right)$ time. Beyond making just point estimates, full estimation of the parameter's posterior density allows us to obtain error bars on estimates and can facilitate active learning and experimental design procedures. Finally, we have seen that making predictions and optimal decisions requires taking expectations with respect to the posterior. Whilst for general multivariate densities such expectations can be difficult to compute, for multivariate Gaussian posteriors the required integrals can often be performed analytically.

2.1.2 Factor analysis

Factor Analysis (FA) is an unsupervised, probabilistic, generative model that assumes the observed real-valued N-dimensional data vectors, $\mathbf{v} \in \mathbb{R}^N$, are Gaussian distributed and can be approximated as lying on some low dimensional linear subspace. Under these assumptions, the model can capture the low dimensional correlational structure of high dimensional data vectors. As such it is used widely throughout machine learning and statistics, both in its own right, for example as a method to detect anomalous data points [Wu and Zhang, 2006], or as a subcomponent of a more complex probabilistic model [Ghahramani and Hinton, 1996]. The FA model assumes that an observed data vector, $\mathbf{v} \in \mathbb{R}^N$, is generated according to

$$\mathbf{v} = \mathbf{F}\mathbf{w} + \boldsymbol{\epsilon}, \tag{2.1.9}$$

where $\mathbf{w} \in \mathbb{R}^D$ is the lower dimensional 'latent' or 'hidden' representation of the data where we assume $\mathbf{w} \sim \mathcal{N}\left(\mathbf{0}, \mathbf{I}\right)$, $\mathbf{F} \in \mathbb{R}^{N\times D}$ is the 'factor loading' matrix describing the linear mapping between the 'latent' and 'visible' spaces, and $\boldsymbol{\epsilon}$ is independent additive Gaussian noise $\boldsymbol{\epsilon} \sim \mathcal{N}\left(\mathbf{0}, \boldsymbol{\Psi}\right)$ with $\boldsymbol{\Psi} = \mathrm{diag}\left(\left[\psi_1, ..., \psi_N\right]\right)$. For the special case of isotropic noise, $\boldsymbol{\Psi} = \psi^2\mathbf{I}$, equation (2.1.9) describes the probabilistic generalisation of the Principal Components Analysis (PCA) model [Tipping and Bishop, 1999].

In this section we consider the FA model under the simplifying assumption that the data has zero mean. Extending the FA model to the non-zero mean setting is trivial – for derivations including non-zero mean estimation we point the interested reader to [Barber, 2012, chapter 21].

For a Bayesian approach to the FA model, the parameters $\mathbf{F}$ and $\boldsymbol{\Psi}$ should be treated as random variables and priors should be specified on them. See Figure 2.1.2 for the graphical model representation of the FA model. Full Bayesian inference would then require estimating the posterior density of $\mathbf{F}, \boldsymbol{\Psi}$

conditioned on a data $\mathcal{D} = \{\mathbf{v}_m\}_{m=1}^M$: $p(\mathbf{F}, \mathbf{\Psi}|\mathcal{D})$. However, computing the posterior, or marginals of it, is in general analytically intractable in this setting (see Minka [2000] for one approach to perform deterministic approximate inference in this model). In this section we consider the simpler task of maximum likelihood estimation of $\mathbf{F}, \mathbf{\Psi}$, showing how $\log p(\mathcal{D}|\mathbf{F}, \mathbf{\Psi})$ can be evaluated and optimised. The presentation can be easily extended to maximum a posteriori estimation by adding the prior densities to the log-likelihood and optimising $\log p(\mathcal{D}|\mathbf{F}, \mathbf{\Psi}) + \log p(\mathbf{F}) + \log p(\mathbf{\Psi})$.

Likelihood

The likelihood of the visible variables $\mathbf{v}$ is defined by marginalising out the hidden variables from the joint specification of the probabilistic model,

$$\begin{aligned} p(\mathbf{v}|\mathbf{F}, \mathbf{\Psi}) &= \int \prod_{n=1}^{N} \mathcal{N}\left(v_n|\mathbf{f}_n^\mathsf{T}\mathbf{w}, \psi_n\right) \mathcal{N}(\mathbf{w}|\mathbf{0}, \mathbf{I})\, d\mathbf{w} \\ &= \int \mathcal{N}(\mathbf{v}|\mathbf{F}\mathbf{w}, \mathbf{\Psi})\, \mathcal{N}(\mathbf{w}|\mathbf{0}, \mathbf{I})\, d\mathbf{w} \\ &= \mathcal{N}\left(\mathbf{v}|\mathbf{0}, \mathbf{F}\mathbf{F}^\mathsf{T} + \mathbf{\Psi}\right), \end{aligned} \tag{2.1.10}$$

where v_n is the n^{th} element of the vector $\mathbf{v}$. The last equality above is obtained from Gaussian marginalisation on the joint density of the visible and hidden variables – see Appendix A.2 for the multivariate Gaussian inference identities required to derive this. Equation (2.1.10) shows us that the FA density is a multivariate Gaussian with a particular constrained form of covariance: $\mathrm{cov}(\mathbf{v}) = \mathbf{F}\mathbf{F}^\mathsf{T} + \mathbf{\Psi}$.

Typically $N \gg D$ and so the symmetric positive definite matrix $\mathbf{\Psi} + \mathbf{F}\mathbf{F}^\mathsf{T}$ requires many fewer parameters than a full unstructured covariance matrix. Exactly $D(N+1)$ parameters define $\mathbf{\Psi} + \mathbf{F}\mathbf{F}^\mathsf{T}$ whereas an unstructured covariance has $\frac{1}{2}N(N+1)$ unique parameters. We might hope then that the FA model will provide a more robust estimate of the covariance of $\mathbf{v}$ than directly estimating its unstructured covariance matrix. Computing the likelihood in the FA model is typically cheaper than computing a general unstructured Gaussian density on $\mathbf{v}$: evaluating the density $\mathcal{N}(\mathbf{v}|\mathbf{0}, \mathbf{\Sigma})$ for a general unstructured covariance $\mathbf{\Sigma} \in \mathbb{R}^{N\times N}$ will scale cubically in N, whereas for the FA model evaluating $\mathcal{N}\left(\mathbf{v}|\mathbf{0}, \mathbf{F}\mathbf{F}^\mathsf{T} + \mathbf{\Psi}\right)$ will scale $O\left(ND^2\right)$.

Given a dataset $\mathcal{D} = \{\mathbf{v}_m\}_{m=1}^M$, and assuming the data points are independent and identically distributed given the parameters of the model, the log-likelihood of the dataset is given by

$$\log p(\mathcal{D}|\mathbf{F}, \mathbf{\Psi}) = \sum_{m=1}^{M} \log \mathcal{N}\left(\mathbf{v}_m|\mathbf{0}, \mathbf{F}\mathbf{F}^\mathsf{T} + \mathbf{\Psi}\right). \tag{2.1.11}$$

Inference

In the FA model a typical inferential task is to calculate the probability of a data point $\mathbf{v}$ conditioned on the model. For example in a novelty detection task, given a test point $\mathbf{v}^*$ we may classify it as 'novel' if its probability is below some threshold.

The FA model is also often used for missing data imputation. For example having observed some subset of the visible variables $\mathbf{v}_I$, with I an index set such that $I \subset \{1, ..., N\}$, we may wish to infer the density of the remaining variables $\mathbf{v}_{\setminus I}$ or some subset of them; that is we want to infer the density

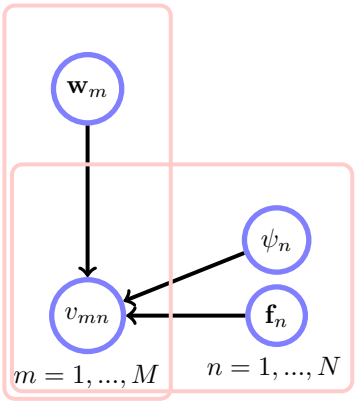


Figure 2.4: Graphical model representation of the factor analysis model. The n^{th} element of the m^{th} observed data point, v_{mn}, is defined by the likelihood $p(v_{mn}|\mathbf{f}_n, \mathbf{w}_m, \psi_n) = \mathcal{N}\left(v_{mn}|\mathbf{f}_n^\mathsf{T}\mathbf{w}_m, \psi_n\right)$. The M latent variables $\mathbf{w}_m \in \mathbb{R}^D$ are assumed Gaussian distributed such that $\mathbf{w}_m \sim \mathcal{N}(\mathbf{0}, \mathbf{I}_D)$. The N factor loading vectors $\mathbf{f}_n$ and noise variances ψ_n are parameters of the model with factorising prior densities $p(\mathbf{F}) = \prod_n p(\mathbf{f}_n)$ and $p(\mathbf{\Psi}) = \prod_n p(\psi_n)$.

$p(\mathbf{v}_{\setminus I}|\mathbf{v}_I, \mathbf{F}, \mathbf{\Psi})$. Due to the bipartite structure of the hidden and latent variables in the FA model, see Figure 2.5, this density can be evaluated by computing

$$p(\mathbf{v}_{\setminus I}|\mathbf{v}_I, \mathbf{F}, \mathbf{\Psi}) = \int \prod_{i \notin I} \mathcal{N}\left(v_i|\mathbf{f}_i^\mathsf{T}\mathbf{w}, \psi_i\right) p(\mathbf{w}|\mathbf{v}_I, \mathbf{F}, \mathbf{\Psi}) d\mathbf{w},$$

where the density $p(\mathbf{w}|\mathbf{v}_I, \mathbf{F}, \mathbf{\Psi})$ is obtained from Bayes' rule

$$p(\mathbf{w}|\mathbf{v}_I, \mathbf{F}, \mathbf{\Psi}) \propto \mathcal{N}(\mathbf{w}|\mathbf{0}, \mathbf{I}) \prod_{i \in I} \mathcal{N}\left(v_i|\mathbf{f}_i^\mathsf{T}\mathbf{w}, \psi_i\right), \tag{2.1.12}$$

since $p(\mathbf{w}|\mathbf{v}_I, \mathbf{F}, \mathbf{\Psi})$ above is defined as the product of two Gaussian densities it is also a Gaussian density whose moments can be computed using the results presented in Appendix A.2. Similarly, the density $p(\mathbf{v}_{\setminus I}|\mathbf{v}_I, \mathbf{F}, \mathbf{\Psi})$ is also Gaussian whose moments can be easily evaluated.

Parameter estimation

Two general techniques to perform parameter estimation in latent variable models are the expectation maximisation algorithm [Dempster et al., 1977] and a gradient ascent procedure using a specific identity for the derivative of the log-likelihood. Both procedures are explained at greater length in Appendix A.3. Neither the EM algorithm nor the gradient ascent procedure are the most efficient parameter estimation techniques for the FA model, for example see Zhao et al. [2008] for a more efficient eigen-based approach. However, we present the EM and gradient ascent procedures since they can be easily adapted to the non-Gaussian linear latent variable models we consider later in this chapter. Since there are many similarities between the EM and gradient ascent procedures we present only the EM method here and leave a discussion of the gradient ascent procedure to the appendix.

Applying the EM algorithm to the FA model, the E-step requires the evaluation of the conditional densities $q(\mathbf{w}_m) = p(\mathbf{w}_m|\mathbf{v}_m, \mathbf{F}, \mathbf{\Psi})$, for each $m = 1, \ldots, M$. Since the FA model is jointly Gaussian on all the random variables, this conditional density is also Gaussian distributed. Applying the Gaussian inference results presented in Appendix A.2.3, each of these densities is given by

$$p(\mathbf{w}_m|\mathbf{v}_m, \mathbf{F}, \mathbf{\Psi}) = \mathcal{N}(\mathbf{w}_m|\mathbf{m}_m, \mathbf{S}),$$

where the moments $\mathbf{m}_m \in \mathbb{R}^D$ and $\mathbf{S} \in \mathbb{R}^{D \times D}$ are defined as

$$\mathbf{S} = \left(\mathbf{F}^\mathsf{T}\mathbf{\Psi}^{-1}\mathbf{F} + \mathbf{I}_D\right)^{-1}, \quad \text{and} \quad \mathbf{m}_m = \mathbf{S}\mathbf{F}^\mathsf{T}\mathbf{\Psi}^{-1}\mathbf{v}_m.$$

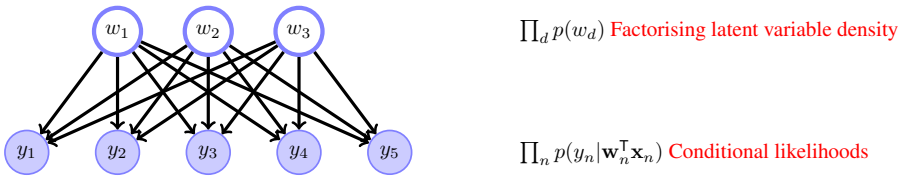


Figure 2.5: Bipartite graphical model structure for a general unsupervised factor analysis model.

Since in the FA model we typically assume $D \ll N$, computing all these conditionals scales as $O\left(MND^2\right)$. Optimising the likelihood using the gradient ascent procedure discussed in Appendix A.3 requires the evaluation of each of these densities for a single evaluation of the derivative of the data log-likelihood.

The M-step of the EM algorithm corresponds to optimising the energy's contribution to the bound on the log-likelihood with respect to the parameters of the model. For the FA model the M-step corresponds to optimising the energy function

$$E(\mathbf{F}, \mathbf{\Psi}) := \sum_{m=1}^{M} \left\langle \log \mathcal{N}\left(\mathbf{v}_m | \mathbf{F}\mathbf{w}_m, \mathbf{\Psi}\right) \right\rangle_{q(\mathbf{w}_m)},$$

with respect to $\mathbf{F}, \mathbf{\Psi}$. Closed form updates can be derived to maximise $E(\mathbf{F}, \mathbf{\Psi})$, and correspond to setting

$$\mathbf{F} = \mathbf{A}\mathbf{H}^{-1},$$
$$\mathbf{\Psi} = \operatorname{diag}\left(\frac{1}{M}\sum_{m=1}^{M} \mathbf{v}_m\mathbf{v}_m^\mathsf{T} - 2\mathbf{F}\mathbf{A}^\mathsf{T} + \mathbf{F}\mathbf{H}\mathbf{F}\right),$$

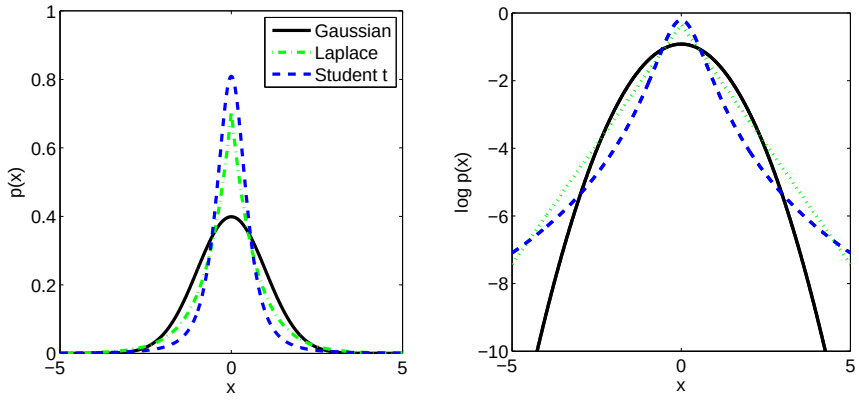
where $\mathbf{H} := \mathbf{S} + \frac{1}{M}\sum_{m=1}^{M} \mathbf{m}_m\mathbf{m}_m^\mathsf{T}$ and $\mathbf{A} := \frac{1}{M}\sum_{m=1}^{M} \mathbf{v}_m\mathbf{m}_m^\mathsf{T}$ – see [Barber, 2012, Section 21.2.2] for a full derivation of this result.

Summary

Factor analysis and probabilistic principal components analysis are simple and widely used models for capturing low dimensional structure in real-valued data vectors. Inference and parameter estimation in the model is facilitated by the Gaussian conjugacy of the latent variable density, $p(\mathbf{w}) = \mathcal{N}\left(\mathbf{w}|\mathbf{0}, \mathbf{I}\right)$, and the conditional likelihood density, $p(\mathbf{v}|\mathbf{w}, \mathbf{F}, \mathbf{\Psi}) = \mathcal{N}\left(\mathbf{y}|\mathbf{F}\mathbf{w}, \mathbf{\Psi}\right)$. The diagonal plus low-rank structure of the Gaussian likelihood covariance matrix provides computational time and memory savings over general unstructured multivariate Gaussian densities. Parameter estimation in the FA model can be implemented by the expectation maximisation algorithm or log-likelihood gradient ascent procedures, both of which require the repeated evaluation of the latent variable conditional densities $\{p(\mathbf{w}_m|\mathbf{v}_m, \mathbf{F}, \mathbf{\Psi})\}_{m=1}^{M}$.

2.2 Latent linear models : approximate inference

In Section 2.1.1 we considered the latent linear model for supervised conditional density estimation in the form of the Bayesian linear regression model. In Section 2.1.2 we considered the latent linear model



(a) (b)

Figure 2.6: Gaussian, Laplace and Student's t densities with unit variance: (a) probability density functions and (b) log probability density functions. Laplace and Student's t densities have stronger peaks and heavier tails than the Gaussian. Student's t with d.o.f. $\nu = 2.5$ and scale $\sigma^2 = 0.2$, Laplace with $\tau = 1/\sqrt{2}$.

for unsupervised density estimation in the form of the factor analysis model. In both cases the Gaussian density assumptions resulted in analytically tractable inference procedures. Furthermore, the resulting Gaussian conditional densities on the latent variables/parameters were also seen to make downstream processing tasks such as forecasting, utility optimisation and parameter optimisation tractable as well.

Whilst computationally advantageous, in both the regression and factor analysis setting, we would often like to extend these models to fit non-Gaussian data. In this section we introduce extensions to the latent linear model class in order to more accurately represent non-Gaussian data.

2.2.1 Non-Gaussian Bayesian regression models

The Bayesian linear regression model presented in Section 2.1.1 can be extended by considering non-Gaussian priors and/or non-Gaussian likelihoods.

Non-Gaussian priors

Conjugacy for the Bayesian linear regression model in Section 2.1.1 was obtained by assuming that the prior $p(\mathbf{w})$ was Gaussian distributed $p(\mathbf{w}) = \mathcal{N}(\mathbf{w}|\boldsymbol{\mu}, \boldsymbol{\Sigma})$. In many settings this assumption may be inaccurate resulting in poor models of the data. For example we may only know, a priori, that the parameters are bounded such that $w_d \in [l_d, u_d]$, in which case a factorising uniform prior would be more appropriate than the Gaussian. Alternatively, in some settings we may believe that only a small subset of the parameters are responsible for generating the data; such knowledge can be encoded by a 'sparse prior' such as a factorising Laplace density or a 'spike and slab' density constructed as a mixture of a Gaussian and a delta 'spike' function at zero. Non-Gaussian, factorising priors and a Gaussian observation noise model describe a posterior of the form

$$p(\mathbf{w}|\mathbf{y}, \mathbf{X}, s) = \frac{1}{Z}\mathcal{N}\left(\mathbf{y}|\mathbf{X}^{\mathsf{T}}\mathbf{w}, s^2\mathbf{I}_N\right)\prod_{d=1}^{D} p(w_d),$$

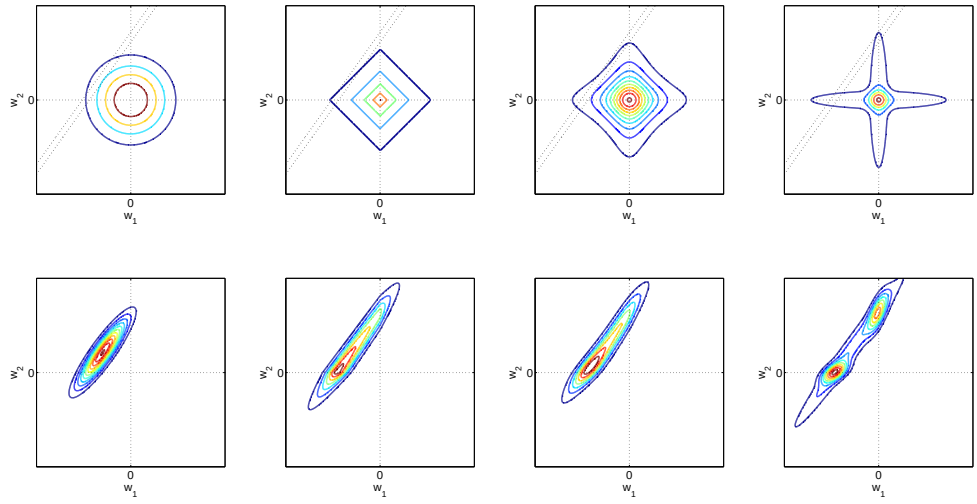


Figure 2.7: Isocontours for a selection of linear model prior, likelihood and resulting posterior densities. The top row plots contours of the two dimensional prior (solid line) and the Gaussian likelihood (dotted line). The second row displays the contours of the posterior induced by the prior and likelihood above it. Column 1 - a Gaussian prior, Column 2 - a Laplace prior, Column 3 - a Student's t prior and Column 4 a spike and slab prior constructed as a product over dimensions of univariate two component Gaussian mixture densities.

where $p(w_d)$ are the independent factors of the non-Gaussian prior. The marginal likelihood, Z in the equation above, and thus also the posterior, typically cannot be computed when $D \gg 1$. Figure (2.7) plots the likelihood, prior and corresponding posterior density contours of a selection of toy two dimensional Bayesian linear regression models with non-conjugate, sparse priors. In Appendix A.5 we provide parametric forms for all the Bayesian linear model priors we use in this thesis.

Non-Gaussian likelihoods

We may also wish to model dependent variables y which cannot be accurately represented by conditional Gaussians. For instance, in many settings the conditional statistics of real-valued dependent variables, y, may be more accurately described by heavy tailed densities such as the Student's t or the Laplace – see Figure 2.6 for a depiction of these density functions. A more significant departure from the model considered in Section 2.1.1 is where the dependent variable is discrete valued, such as for binary $y \in \{-1, +1\}$, categorical $y \in \{1, \ldots, K\}$, ordinal $y \in \{1, \ldots, K\}$ with a fixed ordering, or count dependent variables $y \in \mathbb{N}$. Whilst each of these data categories have likelihoods that can be quite naturally parameterised by a conditional distribution, conjugate priors do not exist. Thus simple analytic forms for the posterior, the marginal likelihood, and the predictive density cannot be derived. Below we consider two popular approaches to extending linear regression models to non-Gaussian dependent variables: the generalised linear model, and the latent response model.

Model	$y \in$	$g : g^{-1}(\mathbf{w}^\mathsf{T}\mathbf{x}) = \mu$	$p(y\|\mu)$	Parameters
Linear regression	$\mathbb{R}$	$g(x) = x$	$\mathcal{N}\left(y\|\mu, \sigma^2\right)$	σ^2
Logistic regression	$\{-1, +1\}$	$g(x) = \log(x) - \log(1-x)$	$\sigma(y\mu)$	$\emptyset$
Poisson regression	$\mathbb{N}$	$g(x) = \log x$	$\frac{\mu^y e^{-\mu}}{y!}$	$\emptyset$

Table 2.1: Some common generalised linear model likelihoods and link functions. Above y is the dependent variable, $\mathbf{x}$ is the covariate vector, $\mathbf{w}$ is the parameter vector and g is the link function defining the mean predictor such that $g^{-1}(\mathbf{w}^\mathsf{T}\mathbf{x}) = \mu$. Additional parameters that are required to specify the conditional distribution are provided in the last column, where the symbol $\emptyset$ denotes the empty set.

Generalised linear models

Generalised Linear Models (GLMs) assume the same conditional dependence structure between the covariates $\mathbf{x}$ and the dependent variables y as the linear regression model but use different conditional distributions to model $p(y|\mathbf{w}, \mathbf{x})$. In a GLM the conditional distribution $p(y|\mathbf{w}, \mathbf{x})$ is in the exponential family and the mean of the dependent variable y is described by the relation $\langle y \rangle = g^{-1}(\mathbf{w}^\mathsf{T}\mathbf{x})$, where the function g is called the link function [McCullagh and Nelder, 1989]. Informally, the link function can be thought of as a means to warp the linear mean predictor $\mathbf{w}^\mathsf{T}\mathbf{x}$ to the domain for which the likelihood's mean parameter is defined.

For example, dependent variables that are binary valued, $y \in \{-1, +1\}$, can be modelled by a GLM with the conditional density a Bernoulli such that $p(y|\mu) = \mu^{\mathbb{I}[y=+1]}(1-\mu)^{\mathbb{I}[y=-1]}$, with mean μ and where $\mathbb{I}[\cdot]$ denotes the indicator function equal to one when its argument is true and zero otherwise. The most commonly used link function for this model is the logit transform $g(x) = \log(x) - \log(1-x)$, the inverse of which is $g^{-1}(x) = e^x/(1+e^x)$. Substituting the inverse link mean function $g^{-1}(\mathbf{w}^\mathsf{T}\mathbf{x})$ into the Bernoulli we obtain a conditional distribution for $p(y|\mathbf{w}, \mathbf{x})$ of the form

$$p(y|\mathbf{w}, \mathbf{x}) = \left(\frac{e^{\mathbf{w}^\mathsf{T}\mathbf{x}}}{1+e^{\mathbf{w}^\mathsf{T}\mathbf{x}}}\right)^{\mathbb{I}[y=+1]} \left(1 - \frac{e^{\mathbf{w}^\mathsf{T}\mathbf{x}}}{1+e^{\mathbf{w}^\mathsf{T}\mathbf{x}}}\right)^{\mathbb{I}[y=-1]} = \frac{1}{1+e^{-y\mathbf{w}^\mathsf{T}\mathbf{x}}} =: \sigma(y\mathbf{w}^\mathsf{T}\mathbf{x}),$$

where $\sigma(x)$ is called the logistic sigmoid function. For this likelihood model, with a dataset consisting of N observation pairs $(y_n, \mathbf{x}_n)$, and a Gaussian prior $\mathbf{w} \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$, the posterior of the Bayesian GLM is defined as

$$p(\mathbf{w}|\mathbf{y}, \mathbf{X}, \boldsymbol{\Sigma}) = \frac{1}{Z}\mathcal{N}(\mathbf{w}|\boldsymbol{\mu}, \boldsymbol{\Sigma}) \prod_{n=1}^{N} \sigma(y_n \mathbf{w}^\mathsf{T}\mathbf{x}_n),$$

where again Z denotes the marginal likelihood of the model. Computing Z, and so also the posterior, is not feasible when $N \gg 1$ since no closed form expression for the integral $Z = \int \mathcal{N}(\mathbf{w}|\boldsymbol{\mu}, \boldsymbol{\Sigma}) \prod_n \sigma(y_n \mathbf{w}^\mathsf{T}\mathbf{x}_n) d\mathbf{w}$ exists.

Another example of a GLM likelihood can be used for the regression modelling of count data where $y \in \mathbb{N}$. In this setting the Poisson distribution is a convenient conditional distribution for y. A suitable link function for the Poisson mean parameter is $g(x) = \log(x)$ since $g^{-1}(x) = e^x : \mathbb{R} \to \mathbb{R}^+$. Thus for GLM Poisson regression the likelihood is parameterised as

$$p(y|\mathbf{w}, \mathbf{x}) = \frac{1}{y!} e^{-\exp(\mathbf{w}^\mathsf{T}\mathbf{x})} \left(\exp(\mathbf{w}^\mathsf{T}\mathbf{x})\right)^y.$$

Name	$y \in$	$p(y\|\tilde{y})$	$p(\tilde{y}\|\mathbf{w},\mathbf{x})$	$p(y\|\mathbf{w},\mathbf{x})$
Logistic sigmoid	$\{-1,+1\}$	$\mathbb{I}\left[\mathtt{sgn}(\tilde{y})=y\right]$	Logistic$(\tilde{y}\|\mathbf{w}^\mathsf{T}\mathbf{x},1)$	$\sigma(y\mathbf{w}^\mathsf{T}\mathbf{x})$
Logistic probit	$\{-1,+1\}$	$\mathbb{I}\left[\mathtt{sgn}(\tilde{y})=y\right]$	$\mathcal{N}\left(\tilde{y}\|\mathbf{w}^\mathsf{T}\mathbf{x},1\right)$	$\Phi(y\mathbf{w}^\mathsf{T}\mathbf{x})$
Ordinal	$\{1,\ldots,K\}$	$\mathbb{I}\left[\tilde{y}\in(l_{k-1},l_k)\right]$	$\mathcal{N}\left(\tilde{y}\|\mathbf{w}^\mathsf{T}\mathbf{x},1\right)$	$\Phi(l_k-\mathbf{w}^\mathsf{T}\mathbf{x})-\Phi(l_{k-1}-\mathbf{w}^\mathsf{T}\mathbf{x})$

Table 2.2: Some common discrete latent response model conditional distributions. Above y is the dependent variable that we wish to model, $\tilde{y}$ is the nuisance latent response variable that is marginalised out, $\mathbf{x}$ is the covariate vector and $\mathbf{w}$ is the vector of parameters. The Logistic sigmoid $\sigma(x)$ and Logistic probit $\Phi(x)$ functions are defined in Appendix A.5.

In Table 2.1 we present a few examples of dependent variable data classes, suitable link functions and exponential family likelihoods. For each of these models inference is analytically intractable since simple closed form expressions for the posterior and marginal likelihood do not exist.

Latent response models

Conditional distributions for non-Gaussian y can also be constructed by considering nuisance, latent response variables $\tilde{y}$ which are marginalised out when evaluating likelihoods. This construction is often called a latent utility model [Manski, 1977]. For example, a latent response model for binary $y \in \{-1,+1\}$ could be constructed by defining $p(y|\tilde{y}) = \mathbb{I}\left[\mathtt{sgn}(\tilde{y})=y\right]$ and $p(\tilde{y}|\mathbf{w},\mathbf{x}) = \mathcal{N}\left(\tilde{y}|\mathbf{w}^\mathsf{T}\mathbf{x},1\right)$, where $\mathtt{sgn}(\cdot)$ is the signum function which returns ± 1 matching the sign of its argument: $\mathtt{sgn}(x) = x/\left|x\right|$. On integrating out the nuisance latent variables $\tilde{y}$ the conditional distribution on the observed dependent variables y is given by $p(y|\mathbf{w},\mathbf{x}) = \Phi(y\mathbf{w}^\mathsf{T}\mathbf{x})$ where $\Phi(x) := \int_{-\infty}^{x}\mathcal{N}\left(t|0,1\right)dt$ is the cumulative standard normal distribution. The latent response model construction can be used to describe many other conditional densities – some of these are presented in Table 2.2 [Albert and Chib, 1993].

2.2.2 Non-Gaussian linear latent variable models

Similarly to the regression models considered above, the FA model can also be extended to model non-Gaussian distributed variables. In many contexts real-valued data is observed to have statistical properties that are markedly different from Gaussian random variables. For example the statistics of natural images and sound are frequently observed to have strongly super-Gaussian, sparse or leptokurtic densities [Olshausen and Field, 1996, Bell and Sejnowski, 1996]. Furthermore, on a different track we may wish to model the correlational structure between real, binary, and categorical valued variables [Khan et al., 2010, Tipping, 1999]. The FA model can be extended to model such data by using non-Gaussian conditional likelihoods $p(v_n|\mathbf{f}_n,\mathbf{w})$ and/or non-Gaussian latent variable densities $p(\mathbf{w})$.

Non-Gaussian latent variables

Various models have been proposed in the statistics and machine learning communities that can be interpreted as extending the standard factor analysis model by using non-Gaussian latent variables. For instance, a probabilistic formulation of the independent components analysis (ICA) model can be obtained by assuming that the latent variables $\mathbf{w}$ are drawn from some non-Gaussian (frequently sparse) factorising density $p(\mathbf{w}) = \prod_d p_d(w_d)$. Assuming non-Gaussian latent variables often results in markedly

different learnt factor loading mappings $\mathbf{F}$ than in the Gaussian case and can facilitate tasks such as blind source separation, signal deconvolution and image deblurring – see for example Girolami [2001], Fergus et al. [2006], Lee et al. [1999].

In ICA models the dimensionality of the latent space is often equal to or greater than the observed space, $D \geq N$, the fundamental form of the inference problem remains unchanged however. For example, assuming additive Gaussian noise $\epsilon \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\Psi})$, the likelihood of the ICA model is defined by the integral

$$p(\mathbf{v}|\mathbf{F}, \boldsymbol{\Psi}, \boldsymbol{\theta}) = \int \mathcal{N}(\mathbf{y}|\mathbf{F}\mathbf{w}, \boldsymbol{\Psi}) \prod_{d=1}^{D} p(w_d|\theta_d) d\mathbf{w}, \tag{2.2.1}$$

where the parameters $\boldsymbol{\theta}^{\mathsf{T}} = [\theta_1, \ldots, \theta_D]$ define the non-Gaussian factorising prior. Since conjugacy between the latent density and the Gaussian conditional likelihood is lost, closed form parametric expressions for the likelihood $p(\mathbf{v}|\mathbf{F}, \boldsymbol{\Psi}, \boldsymbol{\theta})$ typically cannot be derived. Furthermore, the density of the latent variables conditioned on the visible variables,

$$p(\mathbf{w}|\mathbf{v}, \mathbf{F}, \boldsymbol{\Psi}, \boldsymbol{\theta}) = \frac{1}{Z} \mathcal{N}(\mathbf{v}|\mathbf{F}\mathbf{w}, \boldsymbol{\Psi}) \prod_{d=1}^{D} p(w_d|\theta_d), \tag{2.2.2}$$

which is required to optimise parameters using either the expectation maximisation algorithm or log-likelihood gradient ascent procedures, is intractable since Z, equal to the likelihood expressed in equation (2.2.1), cannot be efficiently computed. Even if the normalisation constant Z in equation (2.2.2) were known, efficient parameter optimisation procedures may not be easy to derive since the expectations defined in the energy's contribution to the EM log-likelihood bound may not admit compact analytic forms amenable to optimisation with respect to the parameters $\mathbf{F}, \boldsymbol{\Psi}, \boldsymbol{\theta}$.

Non-Gaussian conditional likelihoods

A further extension to the FA model considered above is to model discrete and/or continuous valued data. So called mixed data factor analysis extends the FA model to capture (low dimensional) correlational structure for data vectors $\mathbf{v}$ whose elements can be either real or discrete random variables [Tipping, 1999, Khan et al., 2010]. Conditional distributions on discrete variables, such as binary or ordinal variables, can be modelled using either the GLM or the latent response model likelihood constructions considered in Section 2.2.1. In either case, assuming a factorising Gaussian density on the latent variables $\mathbf{w} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}_D)$, the likelihood of a mixed data visible variable $\mathbf{v}$ will be defined by the integral

$$p(\mathbf{v}|\mathbf{F}, \boldsymbol{\theta}) = \int \mathcal{N}(\mathbf{w}|\mathbf{0}, \mathbf{I}_D) \prod_{n=1}^{N} p(v_n|\mathbf{f}_n^{\mathsf{T}}\mathbf{w}, \theta_n) d\mathbf{w}, \tag{2.2.3}$$

where $p(v_n|\mathbf{f}_n^{\mathsf{T}}\mathbf{w}, \theta)$ is a conditional density suitable to model v_n's data type and $\boldsymbol{\theta}^{\mathsf{T}} = [\theta_1, \ldots, \theta_N]$. Since again conjugacy has been lost, equation (2.2.3) above is intractable, as is the conditional $p(\mathbf{w}|\mathbf{v}, \mathbf{F}, \boldsymbol{\theta})$, since its normalisation constant Z is equal to the likelihood defined in equation (2.2.3).

Approximate expectation maximisation

In this subsection we briefly consider the task of performing parameter estimation in a general unsupervised latent linear model where, either or possibly both, the latent and conditional distributions are

non-Gaussian. We denote the distribution on the latent or hidden variables as $p(\mathbf{w}|\boldsymbol{\theta}^h) = \prod_d p(w_d|\theta_d^h)$ and the conditional likelihood $p(\mathbf{v}|\mathbf{F}, \mathbf{w}, \boldsymbol{\theta}^v) = \prod_n p(v_n|\mathbf{w}^\mathsf{T}\mathbf{f}_n, \theta_n^v)$ and define $\boldsymbol{\theta}^\mathsf{T} := [\boldsymbol{\theta}^{v\mathsf{T}}, \boldsymbol{\theta}^{h\mathsf{T}}]$. Adapting the presentation made in Appendix A.3, a lower-bound on the log-likelihood, $\log p(\mathcal{D}|\mathbf{F}, \boldsymbol{\theta})$, can be obtained by considering the KL divergence between a variational density $q(\mathbf{w}_m)$ and the model's conditional density $p(\mathbf{w}_m|\mathbf{v}_m, \mathbf{F}, \boldsymbol{\theta})$ for each data point $\mathbf{v}_m$ in the dataset $\mathcal{D} = \{\mathbf{v}_m\}_{m=1}^M$. This lower bound on the log-likelihood of the data can be written

$$\log p(\mathcal{D}|\mathbf{F}, \boldsymbol{\theta}) \geq \sum_{m=1}^{M} \Bigg\{ H[q(\mathbf{w}_m)] + \Big\langle \log p(\mathbf{w}_m|\boldsymbol{\theta}^h) \Big\rangle_{q(\mathbf{w}_m)} + \sum_{n=1}^{N} \Big\langle \log p(v_{mn}|\mathbf{w}^\mathsf{T}\mathbf{f}_n, \theta_n^v) \Big\rangle_{q(\mathbf{w}_m)} \Bigg\}. \quad (2.2.4)$$

The E-step of the exact EM algorithm corresponds to updating the set of variational densities so that $q(\mathbf{w}_m) = p(\mathbf{w}_m|\mathbf{v}_m, \mathbf{F}, \boldsymbol{\theta})$ for each $m = 1, \ldots, M$. To extend the EM algorithm to a model where the densities $\{p(\mathbf{w}_m|\mathbf{v}_m, \mathbf{F}, \boldsymbol{\theta})\}_{m=1}^M$ cannot be inferred exactly in the E-step, one approach is simply to use the best approximation $q(\mathbf{w}_m)$ we can find. We refer to this procedure, where the E-step is inexact, as the approximate EM algorithm. If each approximation $q(\mathbf{w}_m)$ is found from optimising the KL bound on the log-likelihood, equation (2.2.4) for the generalised FA model, the approximate EM algorithm is guaranteed to increase the lower-bound on the log-likelihood but is not guaranteed to increase the log-likelihood itself. However, this procedure is frequently observed to obtain good solutions. If each approximation $q(\mathbf{w}_m)$ is found using some other (non lower-bounding) approximation method, for example the Laplace or the expectation propagation approximations (methods that we discuss in the following chapter), the approximate EM algorithm is not guaranteed to increase the likelihood or a bound on it.

For the approximate EM procedure to be feasible we require that the variational approximate densities, $\{q(\mathbf{w}_m)\}$, can be efficiently computed, and the expectations in the energy's contribution to the KL bound, $\sum_m \langle \log p(\mathbf{v}_m, \mathbf{w}_m|\boldsymbol{\theta}) \rangle_{q(\mathbf{w}_m)}$, can be efficiently optimised.

2.2.3 Summary

As we have seen above both the Bayesian linear regression model and the unsupervised factor analysis model can be easily extended to model non-Gaussian distributed data. The models can be extended by considering both non-Gaussian latent variable densities or priors, $p(\mathbf{w})$, and non-Gaussian conditional likelihoods $p(y|\mathbf{w}, \mathbf{x})$ or $p(v|\mathbf{f}, \mathbf{w})$. Since the conditional dependence structure of these extensions is unchanged compared to the fully Gaussian case, the definitions of the core inferential quantities of interest remain the same. However, since conjugacy between the latent/prior densities and the conditional visible/likelihood densities is lost, analytic closed-form expressions for marginals and conditionals cannot be derived. Since extending the linear model to handle such non-Gaussian data is of significant practical utility we require efficient and accurate methods to approximate these quantities. In the next section we define the general form of the inference problem that this class of models poses.

2.3 Approximate inference problem

For a vector of parameters $\mathbf{w} \in \mathbb{R}^D$, a multivariate Gaussian potential $\mathcal{N}(\mathbf{w}|\boldsymbol{\mu}, \boldsymbol{\Sigma})$ with mean $\boldsymbol{\mu} \in \mathbb{R}^D$ and symmetric positive definite covariance $\boldsymbol{\Sigma} \in \mathbb{R}^{D\times D}$, we want to approximate the density defined as

$$p(\mathbf{w}) = \frac{1}{Z}\mathcal{N}(\mathbf{w}|\boldsymbol{\mu}, \boldsymbol{\Sigma}) \prod_{n=1}^{N} \phi(\mathbf{w}^\mathsf{T}\mathbf{h}_n), \tag{2.3.1}$$

and its normalisation constant Z defined as

$$Z = \int \mathcal{N}(\mathbf{w}|\boldsymbol{\mu}, \boldsymbol{\Sigma}) \prod_{n=1}^{N} \phi_n(\mathbf{w}^\mathsf{T}\mathbf{h}_n) d\mathbf{w}, \tag{2.3.2}$$

where $\phi_n : \mathbb{R} \to \mathbb{R}^+$ are non-Gaussian, real-valued, positive potential functions and $\mathbf{h}_n \in \mathbb{R}^D$ are fixed real-valued vectors. We refer to the individual factors $\phi_n(\mathbf{w}^\mathsf{T}\mathbf{h}_n)$ as site-projection potentials. We call these factors potentials and not densities since they do not necessarily normalise to 1.

As we saw in Section 2.2 estimating equation (2.3.1) and equation (2.3.2) are the core inferential tasks in both Bayesian supervised linear models and unsupervised latent linear models. Typically neither of these quantities can be efficiently computed in problems of even moderate dimensionality – for example $D, N > 10$. Approximations are thus required. In what follows we refer to $p(\mathbf{w})$ as the target density and Z as the normalisation constant.

We note here that the inference problem posed above is that of estimating a joint density $p(\mathbf{w})$ not just its marginals $p(w_d)$ for which other special purpose methods can be derived [Rue et al., 2009, Cseke and Heskes, 2010, 2011]. A further point of note is that we consider inference for general vectors $\{\mathbf{h}_n\}_{n=1}^N$ for which the graph describing the dependence relations on $\mathbf{w}$ is densely connected. That is, we do not consider special cases where $p(\mathbf{w})$ can be expressed in some other structured factorised form which can be used to simplify the inference problem.

Considerations

In approximating the target $p(\mathbf{w})$ and its normalisation constant Z as defined above we would like any approximate inference algorithm to possess the following properties:

- To be accurate. We want that the approximation to Z and $p(\mathbf{w})$ is as good as possible.
- To be efficient. We want the approximate inference method to be fast and scalable. How the complexity of inference scales both with the number of potential functions N and the dimensionality of the parameter space D is important.
- To be generally applicable. We want the method to place few restrictions on the functional form of the site-projection potentials $\{\phi_n(x)\}_{n=1}^N$ to which the method can successfully be applied.

The above three desiderata are the core axes by which we will measure approximate inference methods. Clearly, trade offs between these criteria will have to be made. In the next chapter we provide an introduction and overview of the deterministic approximate inference methods commonly applied to latent linear models of the form introduced above and consider how they perform against these criteria.

Chapter 3

Deterministic approximate inference

In this chapter we introduce and review some of the most commonly used deterministic approximate inference methods that are applied to the latent linear model class. First we consider the Kullback-Leibler divergence measure which is used to drive and evaluate many of these methods. Then we consider each of the deterministic approximate inference methods in turn assessing the approximation algorithms in terms of their accuracy, their efficiency and the range of models to which they can be applied. Specifically we consider: the MAP approximation, mean field bounding methods, the Laplace approximation, the Gaussian expectation propagation approximation, Gaussian Kullback-Leibler bounding methods and local variational bounding methods. Finally, in Section 3.10 we go on to discuss some extensions to these methods that have been proposed to increase the accuracy of their approximations.

3.1 Approximate inference

As we saw in Section 2.1, probabilistic inference, at its core, is a numerical integration problem. When the dimensionality of the integral is too large and the integral does not have a simple analytic form approximations are required. The approximation methods commonly applied to this integration problem fall into two, broadly distinct, categories: sampling based methods such as Monte Carlo Markov chain and deterministic variational methods. The focus of this thesis is the latter approach.

Sampling methods for approximate Bayesian inference are the subject of a broad and deep literature in machine learning and statistics and so evaluating these techniques is beyond the scope of this thesis. For an introduction to these methods from the perspective of machine learning applications we point the interested reader to Andrieu et al. [2003] and references therein for details. However, we briefly note that sampling methods are generally applicable and, given enough computational resource, one of the most accurate approaches to approximating posterior densities. Whilst sampling based methods can be extremely accurate they are also typically slow to converge to accurate solutions, with convergence itself a property that is difficult to diagnose [Cowles and Carlin, 1996]. What is more, whilst there are many techniques to generate samples from the posterior target density, approximating the normalisation constant, Z in equation (2.3.2), is typically a much harder task using sampling methods. Thus sampling techniques are not well suited to empirical Bayesian model selection techniques such as the ML-II procedure discussed in Section 2.1.1.

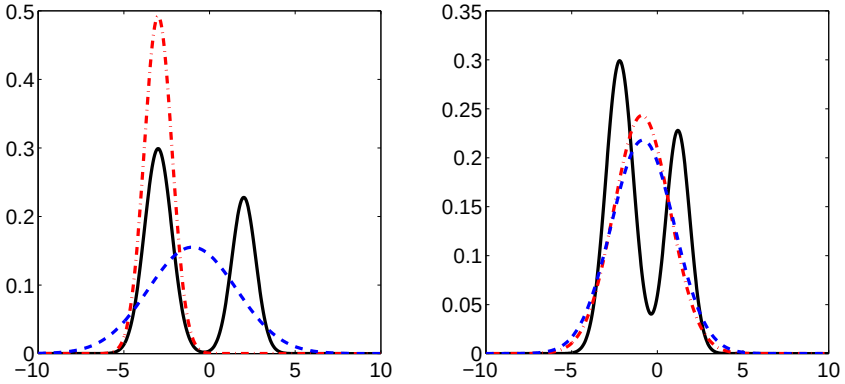


Figure 3.1: A two component Gaussian mixture target density $p(w)$ (solid black line) and the closest univariate Gaussian approximation $q(w)$ found from minimising either $\mathrm{KL}(q(w)|p(w))$ (red dash-dotted line) or $\mathrm{KL}(p(w)|q(w))$ (blue dashed line). (a) Gaussian mixture target $p(w) = 0.6\mathcal{N}(w|-3, 0.8) + 0.4\mathcal{N}(w|2, 0.7)$. (b) Gaussian mixture target $p(w) := 0.6\mathcal{N}(w|-2.2, 0.8) + 0.4\mathcal{N}(w|1.2, 0.7)$.

Deterministic approximate inference methods seek to approximate the target by a density from some fixed family of simpler distributions. The 'best' or 'closest' approximation to the target is typically found by optimising some measure of the goodness of the approximation. Therefore, deterministic approximate inference methods translate a numerical *integration* problem into a numerical *optimisation* problem.

Deterministic approximate inference methods typically require that the target density satisfies some constrained functional form and so generally are less widely applicable than sampling techniques. Furthermore, since deterministic methods approximate the target by some simpler constrained density, their accuracy is limited. However, when these methods are applicable, they can provide fast and easy to implement routines to approximate both the posterior and its normalisation constant. Ease of implementation, an estimate of Z, and speed, are the core advantages of the deterministic variational approach to approximate inference.

When latent linear models describe a unimodal target density, we can often reasonably expect them to be well approximated by a density from some simple constrained class and so we may not require the representational flexibility that sampling methods provide. Indeed, a significant body of literature in machine learning and statistics justifies the use of deterministic variational approximate inference methods in this model class (for example see [Wainwright and Jordan, 2008, Barber, 2012, Seeger, 2008] and references therein). In this chapter we introduce some of the most commonly used deterministic approximate inference methods applied to latent linear models and assess the general characteristics of these procedures against the desiderata laid out in Section 2.3.

3.2 Divergence measures

Many deterministic approximate inference methods obtain an approximation to the target $p(\mathbf{w})$ by minimising some measure of the discrepancy between $p(\mathbf{w})$ and a simpler approximating 'variational' den-

sity $q(\mathbf{w})$. Equipped with a divergence or distance metric between our approximation $q(\mathbf{w})$ and our target $p(\mathbf{w})$, denoted $D(q(\mathbf{w})|p(\mathbf{w}))$, the problem of approximate inference reduces to the optimisation problem

$$q^*(\mathbf{w}) = \underset{q\in\mathcal{Q}}{\operatorname{argmin}}\, D(p(\mathbf{w})|q(\mathbf{w})), \tag{3.2.1}$$

where $\mathcal{Q}$ denotes the set approximating variational densities that we perform the optimisation over. The larger the set $\mathcal{Q}$ is that we can evaluate and optimise equation (3.2.1) over the better the approximation $q^*(\mathbf{w})$ to $p(\mathbf{w})$ has the potential to be. If $p(\mathbf{w}) \in \mathcal{Q}$ then $q^*(\mathbf{w}) = p(\mathbf{w})$.

Thus before considering specific deterministic inference methods, in this section we first discuss the divergence measures that are commonly used to drive deterministic approximate inference methods. Many approximate inference methods used in machine learning and statistics can be viewed as α-divergence optimisation methods [Minka, 2005]. The α-divergence is defined as

$$D_\alpha(p(\mathbf{w})|q(\mathbf{w})) := \frac{1}{\alpha\,(1-\alpha)}\left(1 - \int p(\mathbf{w})^\alpha q(\mathbf{w})^{1-\alpha} d\mathbf{w}\right),$$

where α is a real-valued parameter of the divergence. The α-divergence is zero, for all values of α, if and only if $p(\mathbf{w}) = q(\mathbf{w})$ almost everywhere, otherwise the α-divergence is positive. Whilst not a true metric then, for example it does not in general satisfy the triangle inequality, the α-divergence is a measure of the discrepancy between the two distributions. Computing the α-divergence, which requires performing a D-dimensional integral (where D is the dimensionality of $\mathbf{w}$), is typically intractable. However, the form of the integrand simplifies considerably at the limits $\alpha \to 0$ and $\alpha \to 1$.

At the limits $\alpha \to 0$ and $\alpha \to 1$, the α-divergence is known as the Kullback-Leibler divergence or the relative entropy – see [Cover and Thomas, 1991] for an introduction. Specifically we have that

$$\lim_{\alpha\to 0} D_\alpha(p(\mathbf{w})|q(\mathbf{w})) = \mathrm{KL}(q(\mathbf{w})|p(\mathbf{w})) \quad \text{and} \quad \lim_{\alpha\to 1} D_\alpha(p(\mathbf{w})|q(\mathbf{w})) = \mathrm{KL}(p(\mathbf{w})|q(\mathbf{w}))\,,$$

where the functional $\mathrm{KL}(q(\mathbf{w})|p(\mathbf{w}))$ is defined as

$$\mathrm{KL}(q(\mathbf{w})|p(\mathbf{w})) := \int_{\mathcal{W}} q(\mathbf{w}) \log \frac{q(\mathbf{w})}{p(\mathbf{w})} d\mathbf{w} = \langle \log q(\mathbf{w}) \rangle_{q(\mathbf{w})} - \langle \log p(\mathbf{w}) \rangle_{q(\mathbf{w})}\,, \tag{3.2.2}$$

and $\mathcal{W}$ is the support of $q(\mathbf{w})$. The KL divergence has the properties: $\mathrm{KL}(q(\mathbf{w})|p(\mathbf{w})) \geq 0$ for all densities $p(\mathbf{w})$, $q(\mathbf{w})$; $\mathrm{KL}(q(\mathbf{w})|p(\mathbf{w})) = 0$ iff $q(\mathbf{w}) = p(\mathbf{w})$ almost everywhere; and $\mathrm{KL}(q(\mathbf{w})|p(\mathbf{w})) \neq \mathrm{KL}(p(\mathbf{w})|q(\mathbf{w}))$ for $q(\mathbf{w}) \neq p(\mathbf{w})$.

As the KL divergence is not symmetric, below we consider the $\mathrm{KL}(q(\mathbf{w})|p(\mathbf{w}))$ and the $\mathrm{KL}(p(\mathbf{w})|q(\mathbf{w}))$ forms separately as approximate inference objective functions.

Kullback-Leibler divergence : $\mathbf{KL}(q(\mathbf{w})|p(\mathbf{w}))$

For $q(\mathbf{w})$ from some constrained distribution class and $p(\mathbf{w})$ the target density as defined in Section 2.3 we have $\mathrm{KL}(q(\mathbf{w})|p(\mathbf{w}))$ defined as

$$\begin{aligned}\mathrm{KL}(q(\mathbf{w})|p(\mathbf{w})) &= \left\langle \log \frac{q(\mathbf{w})}{p(\mathbf{w})} \right\rangle_{q(\mathbf{w})} \\ &= \langle \log q(\mathbf{w}) \rangle - \langle \log \mathcal{N}(\mathbf{w}|\boldsymbol{\mu}, \boldsymbol{\Sigma}) \rangle - \sum_{n=1}^{N} \left\langle \log \phi_n(\mathbf{w}^\mathsf{T}\mathbf{h}_n) \right\rangle + \log Z, \end{aligned} \tag{3.2.3}$$

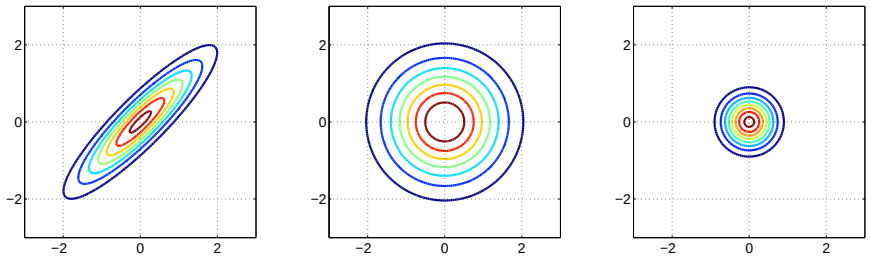


Figure 3.2: (a) Contours of a correlated bivariate Gaussian density target $p(\mathbf{w})$ with zero mean, unit variance and covariance 0.9. (b) Contours of the optimal factorising Gaussian approximation $q(\mathbf{w})$ to $p(\mathbf{w})$ found by minimising the KL divergence $\mathrm{KL}(p(\mathbf{w})|q(\mathbf{w}))$. (c) Contours of the optimal factorising Gaussian approximation $q(\mathbf{w})$ to $p(\mathbf{w})$ found by minimising the KL divergence $\mathrm{KL}(q(\mathbf{w})|p(\mathbf{w}))$. The $\mathrm{KL}(q(\mathbf{w})|p(\mathbf{w}))$ objective results in mode seeking approximations, whilst the $\mathrm{KL}(p(\mathbf{w})|q(\mathbf{w}))$ objective results in support covering approximations.

where the expectations $\langle\cdot\rangle$ in the equation above are taken with respect to the variational density $q(\mathbf{w})$. Variational approximation methods that seek to optimise the objective in equation (3.2.3) are limited by the class of distributions, $\mathcal{Q}$, for which these expectations can be computed: in mean field methods $\mathcal{Q}$ is the set of fully factorising densities [Opper, 2001], in 'variational Bayes' methods $\mathcal{Q}$ is a particular set of block factorising densities [Beal, 2003], in Gaussian KL methods $\mathcal{Q}$ is the set of multivariate Gaussian densities [Barber and Bishop, 1998a, Opper and Archambeau, 2009]. In Chapter 6 we introduce a new method to optimise $\mathrm{KL}(q(\mathbf{w})|p(\mathbf{w}))$ where $q(\mathbf{w})$ is constructed as an affine transformation of a factorising density.

As we can see in equation (3.2.3), one of the core advantages of the $\mathrm{KL}(q(\mathbf{w})|p(\mathbf{w}))$ objective over other α-divergences is that the target density only appears inside the log function. Since the target density $p(\mathbf{w})$ is typically defined as a product over many potentials, taking its logarithm often results in a significant analytic and hence computational simplification. What is more, the logarithm separates the potential functions from the normalisation constant which is an unknown quantity. Thus provided the expectations of equation (3.2.3) can be evaluated the KL divergence can be computed up to the constant $\log Z$.

The KL divergence is non-negative, so rearranging equation (3.2.3) provides a lower-bound on $\log Z$ in the form

$$\log Z \geq \mathcal{B}_{KL} := \underbrace{-\left\langle \log q(\mathbf{w}) \right\rangle}_{\text{entropy}} + \underbrace{\left\langle \log \mathcal{N}\left(\mathbf{w}|\boldsymbol{\mu}, \boldsymbol{\Sigma}\right) \right\rangle + \sum_{n=1}^{N} \left\langle \log \phi_n(\mathbf{w}^{\mathsf{T}}\mathbf{h}_n) \right\rangle}_{\text{energy}}. \tag{3.2.4}$$

As we saw in Section 2.1, the normalisation constant Z is an essential quantity in empirical probabilistic modelling – necessary to compute likelihoods and so perform model selection and parameter estimation. When exact inference is intractable, a lower-bound to the normalisation constant can be used as its surrogate. Lower-bounds have an advantage over approximations in that they provide exact, concrete

knowledge about Z. On a practical note, optimising a lower-bound is typically more numerically stable than approximation methods that do not optimise a numerical objective function.

Having found the optimal approximating density $q^*(\mathbf{w})$ that minimises $\mathrm{KL}(q(\mathbf{w})|p(\mathbf{w}))$, for $q(\mathbf{w}) \in \mathcal{Q}$ some restricted class, what are the general properties of this approximation? Some insight can be gained by inspecting the form of equation (3.2.3). Since $\mathrm{KL}(q(\mathbf{w})|p(\mathbf{w}))$ is the expectation of $\log\left(q(\mathbf{w})/p(\mathbf{w})\right)$ with respect to $q(\mathbf{w})$, $q(\mathbf{w})$ will be forced to be small where $p(\mathbf{w})$ is small [Minka, 2005]. This zero forcing property results in the so called 'mode seeking' behaviour of the $\mathrm{KL}(q(\mathbf{w})|p(\mathbf{w}))$ approximation. The consequence being that $q^*(\mathbf{w})$ will avoid making false positive approximation errors, possibly at the expense of false negatives. Whilst this tendency to underestimate the target density's entropy or variance is commonly observed, it is not guaranteed [MacKay et al., 2008]. See Figure 3.2 and Figure 3.1 for a depiction of the mode seeking behaviour of the $\mathrm{KL}(q(\mathbf{w})|p(\mathbf{w}))$ approximation. In Figure 3.1 a univariate Gaussian $q(w)$ is fitted to a two component Gaussian mixture target density $p(w)$, for the well separated Gaussian mixture target the optimal Gaussian approximation places all its mass at a single mode of the target. Similarly, in Figure 3.2, when a bivariate factorising Gaussian $q(\mathbf{w})$ is fitted to a correlated Gaussian $p(\mathbf{w})$, the optimal variational density covers only the axis aligned support of the target.

Kullback-Leibler divergence : $\mathbf{KL}(p(\mathbf{w})|q(\mathbf{w}))$

The KL divergence $\mathrm{KL}(p(\mathbf{w})|q(\mathbf{w}))$ is given by

$$\mathrm{KL}(p(\mathbf{w})|q(\mathbf{w})) = \left\langle \log \frac{p(\mathbf{w})}{q(\mathbf{w})} \right\rangle_{p(\mathbf{w})} = \langle \log p(\mathbf{w}) \rangle_{p(\mathbf{w})} - \langle \log q(\mathbf{w}) \rangle_{p(\mathbf{w})} , \tag{3.2.5}$$

which requires the evaluation of expectations with respect to the intractable target density $p(\mathbf{w})$. As a result, computing $\mathrm{KL}(p(\mathbf{w})|q(\mathbf{w}))$ is generally intractable and approximations to this objective are required. Expectation propagation, a technique we discuss in Section 3.6, is one method that seeks to approximately optimise this objective [Minka, 2001a].

Whilst equation (3.2.5) is intractable, it is instructive to consider its properties as an objective so as to better understand methods that optimise approximations of it. Considering that $\mathrm{KL}(p(\mathbf{w})|q(\mathbf{w}))$ is the expectation of $\log\left(p(\mathbf{w})/q(\mathbf{w})\right)$ with respect to $p(\mathbf{w})$, we can see that $q(\mathbf{w})$ will be forced to cover the entire support of $p(\mathbf{w})$ and thus avoid making false negative approximation errors. In Figure 3.1 and Figure 3.2 we plot the optimal variational approximation $q^*(\mathbf{w})$ found using the $\mathrm{KL}(p(\mathbf{w})|q(\mathbf{w}))$ divergence as the variational objective. In Figure 3.1, for the univariate Gaussians mixture target, $q^*(w)$ seeks to cover the entire support of $p(w)$. When the mixture components are well separated we have the undesirable result that the mode of $q^*(\mathbf{w})$ is in a region of low density for $p(w)$. Thus when $p(\mathbf{w})$ is multimodal optimising $\mathrm{KL}(p(\mathbf{w})|q(\mathbf{w}))$ for $q(\mathbf{w})$ unimodal may not be sensible.

Further insight into the properties of the $\mathrm{KL}(p(\mathbf{w})|q(\mathbf{w}))$ objective can be gained by considering $q(\mathbf{w})$ in the exponential family. If $q(\mathbf{w})$ is an exponential family distribution its density can be expressed as

$$q(\mathbf{w}) = g(\boldsymbol{\eta}) h(\mathbf{w}) \exp\left(\boldsymbol{\eta}^\mathsf{T} \mathbf{u}(\mathbf{w})\right) , \tag{3.2.6}$$

where $\boldsymbol{\eta}$ is the vector of natural parameters, $\mathbf{u}(\mathbf{w})$ is a vector collecting the sufficient statistics of the density, and $g(\boldsymbol{\eta})$ is a constant such that equation (3.2.6) normalises to one. For $q(\mathbf{w})$ in this form, and on ignoring terms constants with respect to the variational parameters $\boldsymbol{\eta}$, $\mathrm{KL}(p(\mathbf{w})|q(\mathbf{w}))$ simplifies to

$$\mathrm{KL}(p(\mathbf{w})|q(\mathbf{w})) \overset{\text{c.}}{=} -\log g(\boldsymbol{\eta}) - \left\langle \boldsymbol{\eta}^{\mathsf{T}}\mathbf{u}(\mathbf{w}) \right\rangle_{p(\mathbf{w})} .$$

Taking the derivative of the equation above with respect to the variational parameters $\boldsymbol{\eta}$ and equating the derivative to zero we obtain

$$\frac{\partial}{\partial \boldsymbol{\eta}} - \log g(\boldsymbol{\eta}) = \langle \mathbf{u}(\mathbf{w}) \rangle_{p(\mathbf{w})} .$$

However, as we show in Appendix A.4, $\frac{\partial}{\partial \boldsymbol{\eta}} - \log g(\boldsymbol{\eta}) = \langle \mathbf{u}(\mathbf{w}) \rangle_{q(\mathbf{w})}$, and so at this setting of $\boldsymbol{\eta}$ we have that

$$\langle \mathbf{u}(\mathbf{w}) \rangle_{p(\mathbf{w})} = \langle \mathbf{u}(\mathbf{w}) \rangle_{q(\mathbf{w})} .$$

Thus a fixed point of $\mathrm{KL}(p(\mathbf{w})|q(\mathbf{w}))$ with $q(\mathbf{w})$ in the exponential family exists when the expected sufficient statistics between the target density and the variational densities are equal. Whilst generally the expected sufficient statistics $\langle \mathbf{u}(\mathbf{w}) \rangle_{p(\mathbf{w})}$ can not easily be computed, the result above shows us that using the $\mathrm{KL}(p(\mathbf{w})|q(\mathbf{w}))$ objective for approximate inference will result in making a moment-matching approximation.

Relation to Fisher information

The Fisher information is defined as the expectation of the second moments of the log density's derivative. Specifically, for $p(\mathbf{w}|\boldsymbol{\theta})$ a density on $\mathbf{w}$ with parameters $\boldsymbol{\theta}$, the Fisher information is defined as

$$\mathbf{I}_F(\boldsymbol{\theta}) = -\left\langle \left(\frac{\partial}{\partial \boldsymbol{\theta}} \log p(\mathbf{w}|\boldsymbol{\theta}) \right) \left(\frac{\partial}{\partial \boldsymbol{\theta}} \log p(\mathbf{w}|\boldsymbol{\theta}) \right)^{\mathsf{T}} \right\rangle = -\left\langle \frac{\partial^2}{\partial \boldsymbol{\theta} \partial \boldsymbol{\theta}^{\mathsf{T}}} \log p(\mathbf{w}|\boldsymbol{\theta}) \right\rangle , \tag{3.2.7}$$

where the expectation is taken with respect to $p(\mathbf{w}|\boldsymbol{\theta})$. The second equality in equation (3.2.7) holds on $p(\mathbf{w}|\boldsymbol{\theta})$ satisfying certain regularity conditions [Cover and Thomas, 1991]. The Fisher information can be viewed as scoring how sharply peaked $p(\mathbf{w}|\boldsymbol{\theta})$ is about its mode with respect to $\boldsymbol{\theta}$. The Fisher information, by means of the Cramer-Rao bound, provides a lower-bound on the variance any unbiased estimator can achieve.

Whilst the Fisher information is typically used to score the potential for parameter estimation accuracy, it is also asymptotically related to the KL divergence. The Fisher information describes the local curvature of the KL divergence between two densities $p(\mathbf{w}|\boldsymbol{\theta})$ and $q(\mathbf{w}|\boldsymbol{\theta})$ that are perturbations of one another. For $p(\mathbf{w}|\boldsymbol{\theta})$ a perturbation of $q(\mathbf{w}|\boldsymbol{\theta})$ it can be shown that the second order Taylor expansion of the KL divergence about $\boldsymbol{\theta}_0$ is given by

$$\mathrm{KL}(q(\mathbf{w}|\boldsymbol{\theta})|p(\mathbf{w}|\boldsymbol{\theta}_0) \approx \frac{1}{2} (\boldsymbol{\theta} - \boldsymbol{\theta}_0)^{\mathsf{T}} \mathbf{I}_F(\boldsymbol{\theta}_0) (\boldsymbol{\theta} - \boldsymbol{\theta}_0) ,$$

which is symmetric in $q(\mathbf{w}|\boldsymbol{\theta})$ and $p(\mathbf{w}|\boldsymbol{\theta})$ [Gourieroux and Monfort, 1995]. Thus the Fisher information can be interpreted as a local distance measure between densities. This observation has motivated the development of natural gradient preconditioning methods to speed up gradient ascent procedures to optimise the KL variational bound [Honkela et al., 2010, Amari, 1998].

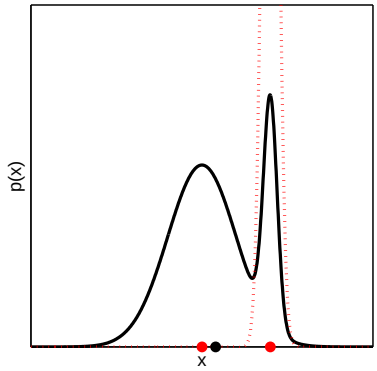


Figure 3.3: Univariate bimodal target density $p(x)$ (black solid line) with mean point (black x-axis dot) and two local maxima (red x-axis dots). The Laplace approximation is centred at the mode of the target density (red dotted).

3.3 MAP approximation

One of the simplest deterministic approximate inference methods is to approximate the target density by a delta function. The maximum a posteriori (MAP) approximation centers the delta function at the mode of the target density $\mathbf{w}_{MAP}$ so that $q(\mathbf{w}) = \delta(\mathbf{w} - \mathbf{w}_{MAP})$. Mode estimation is simpler than mean estimation since it does not require that we perform a multivariate integral over $p(\mathbf{w})$. The MAP estimate can be found by optimising the unnormalised target density since

$$\begin{aligned}\mathbf{w}_{MAP} &:= \underset{\mathbf{w}}{\operatorname{argmax}}\, p(\mathbf{w}) = \underset{\mathbf{w}}{\operatorname{argmax}}\, \log p(\mathbf{w}) \\ &= \underset{\mathbf{w}}{\operatorname{argmax}}\, \log \mathcal{N}(\mathbf{w}|\boldsymbol{\mu}, \boldsymbol{\Sigma}) + \sum_{n=1}^{N} \log \phi_n(\mathbf{w}^\mathsf{T}\mathbf{h}_n),\end{aligned} \tag{3.3.1}$$

where $\log Z$ can be dropped from the objective since it is invariant to $\mathbf{w}$.

The MAP approximation is generally applicable placing few restrictions on the target densities $p(\mathbf{w})$ to which it can be applied. However, the approximation may only be reasonable provided the target is unimodal with negligible variance. Typically this will only be the case provided the site potentials are log-concave and there is sufficient 'data' relative to the dimensionality of the parameter space: $N \gg D$.

Approximation to $p(\mathbf{w})$

Having computed $\mathbf{w}_{MAP}$, the delta approximation $\delta(\mathbf{w} - \mathbf{w}_{MAP})$ can be used as a surrogate to the target $p(\mathbf{w})$ for downstream processing tasks. Since computing an expectation with respect to $q(\mathbf{w}) = \delta(\mathbf{w} - \mathbf{w}_{MAP})$ is equivalent to substituting in $\mathbf{w}_{MAP}$ for $\mathbf{w}$, this approximation makes downstream computations extremely efficient. Indeed, the core advantage of the MAP approximation is the computational saving it offers over other approximate methods.

Approximation to Z

A significant disadvantage of the MAP approximation is that it does not provide an estimate of the normalisation constant Z. Whilst many practitioners use the unnormalised target evaluated at the MAP point, $\mathcal{N}(\mathbf{w}_{MAP}|\boldsymbol{\mu}, \boldsymbol{\Sigma}) + \sum_n \log \phi_n(\mathbf{w}_{MAP}^\mathsf{T}\mathbf{h}_n)$, as an estimate of Z this is not a reliable metric. Substituting the delta approximation as $q(\mathbf{w})$ into the KL bound on $\log Z$, see equation (3.2.4), the bound's entropy contribution will be $-\infty$. Therefore, using $\mathcal{N}(\mathbf{w}_{MAP}|\boldsymbol{\mu}, \boldsymbol{\Sigma}) + \sum_n \log \phi_n(\mathbf{w}_{MAP}^\mathsf{T}\mathbf{h}_n)$

as an approximation to $\log Z$ is unreliable and cannot be used to drive model selection or likelihood maximisation procedures. See Welling et al. [2008] for a deeper discussion of some of the pathologies of this approximation when used for parameter estimation.

Optimisation and complexity

The complexity of the MAP approximation method is equivalent to the complexity of the optimisation problem posed in equation (3.3.1), as such it depends on the functional properties of the target $p(\mathbf{w})$. When all the potential functions $\{\phi_n\}$ are log-concave this is a convex optimisation problem for which many efficient methods have been developed – see for example Nocedal and Wright [2006], Boyd and Vandenberghe [2004]. Provided also that the potentials are twice continuously differentiable quadratic convergence rates can be achieved using Newton's method. If the problem is of sufficiently large dimensionality such that full evaluation of the Hessian is infeasible, quasi-Newton methods such as non-linear conjugate gradients, LBFGS or Hessian free Newton methods can be used. Hessian free Newton methods approximately solve the Newton update using linear conjugate gradients with finite difference approximations for Hessian vector products and can be fast and scalable for convex problems [Nocedal and Wright, 2006].

Storing the MAP approximate posterior requires just D parameters and evaluating the MAP objective typically scales just $O\left(D(D+N)\right)$. The MAP approximation is the cheapest approximate inference method considered here.

Qualities of approximation

The MAP approximation is widely applicable and the computationally cheapest approximation method considered in this chapter. Unfortunately, ignoring measure in parameter space can result in extremely poor approximations: the MAP estimate can be arbitrarily located since it is not invariant to reparameterisations of parameter space, ignoring posterior variance results in overly confident predictions and a reliable estimate of the normalisation constant cannot be derived.

The MAP parameter estimate is not invariant to transformations of parameter space. Invariance would require that $\underset{\mathbf{w}}{\operatorname{argmax}}\, p(\mathbf{w}) = f\left(\underset{\mathbf{v}}{\operatorname{argmax}}\, p\left(\mathbf{v}\right)\right)$ where $\mathbf{w} = f(\mathbf{v})$ and $f : \mathbb{R}^D \to \mathbb{R}^D$ is bijective. The density of the transformed distribution $p(\mathbf{v})$ is defined

$$p(\mathbf{v}) = \int \delta\left(\mathbf{v} - f^{-1}(\mathbf{w})\right) p(\mathbf{w}) d\mathbf{w} = p\left(f(\mathbf{v})\right) \left|\det\left(\frac{\partial f(\mathbf{v})}{\partial \mathbf{v}}\right)\right|,$$

where the last factor in the equation above is the Jacobian of the transformation $f : \mathbf{v} \to \mathbf{w}$. Invariance of the MAP approximation to reparameterisations will only hold then when the Jacobian is invariant to $\mathbf{v}$, for example when the reparameterisation is linear. Thus the MAP point can be arbitrarily located – this is a consequence of the fact that the volume surrounding the modal point can have negligible mass – see Figure 3.3.

Using the delta approximation to $p(\mathbf{w})$ for downstream inferences will typically result in making overly confident predictions. As we saw in Section 2.1 we are required to compute expectations with respect to $p(\mathbf{w})$ to make predictions and optimise parameters. For the Bayesian linear regression model considered in Section 2.1.1, using the MAP approximation to compute the predictive density estimate,

we would make the approximation $\mathcal{N}\left(y_*|\mathbf{m}^\mathsf{T}\mathbf{x}_*, \mathbf{x}_*^\mathsf{T}\mathbf{S}\mathbf{x}_* + s^2\right) \approx \mathcal{N}\left(y_*|\mathbf{w}_{MAP}^\mathsf{T}\mathbf{x}_*, s^2\right)$. Thus the posterior's contribution to the predictive variance, $\mathbf{x}_*^\mathsf{T}\mathbf{S}\mathbf{x}_*$, is ignored using the MAP approximation and so consequently predictions are over confident.

Since the MAP approximation is not found by optimising a bound on the likelihood, using the MAP approximation for the E-step in an approximate EM algorithm is not guaranteed to increase the likelihood or a bound on it.

3.4 Mean field bounding

Mean Field (MF) methods seek to minimise $\mathrm{KL}(q(\mathbf{w})|p(\mathbf{w}))$ over $\mathcal{Q}$ the class of fully factorising approximations $q(\mathbf{w}) = \prod_d q(w_d)$. The KL variational bound, see equation (3.2.4), for this distribution class can be written

$$\log Z \geq \mathcal{B}_{MF} := \sum_{d=1}^{D} H\left[q(w_d)\right] + \left\langle \log \mathcal{N}\left(\mathbf{w}|\boldsymbol{\mu}, \boldsymbol{\Sigma}\right)\right\rangle + \sum_{n=1}^{N} \left\langle \log \phi_n(\mathbf{w}^\mathsf{T}\mathbf{h}_n)\right\rangle, \tag{3.4.1}$$

where the expectations are taken with respect to the factorising variational density $q(\mathbf{w})$. The mean field bound, $\mathcal{B}_{MF}$, is optimised by coordinate ascent in the factors $\{q(w_d)\}_{d=1}^D$. Asynchronously updating the factors is guaranteed to increase the bound. Each factor $q(w_k)$ is updated by taking the functional derivative of equation (3.4.1) with respect to it, equating the derivative to zero and solving subject to normalisation constraints.

For latent linear model target densities, evaluating the mean field bound, equation (3.4.1), and the corresponding factor updates is not always tractable. The expectation of the multivariate Gaussian potential in equation (3.4.1) can typically be evaluated for factorising $q(\mathbf{w})$. The difficulty in applying mean field methods to latent linear models is due to the site-projection potential's contribution to the KL bound $\sum_n \left\langle \log \phi_n(\mathbf{w}^\mathsf{T}\mathbf{h}_n)\right\rangle$. For $\{\phi_n\}$ non-Gaussian, the MF bound is typically intractable for general $\mathbf{H} := [\mathbf{h}_1, ..., \mathbf{h}_N]$. However, if the 'data' vectors are standard normal basis vectors, *i.e.* $\mathbf{H} = \mathbf{I}_D$, and so each factor depends on only a single element of the parameter vector, $\prod_n \phi_n(\mathbf{w}^\mathsf{T}\mathbf{h}_n) = \prod_d \phi_d(w_d)$, the site potentials expectations simplify to $\langle \log \phi_d(w_d)\rangle_{q(w_d)}$. As a consequence the bound and the factor update equations can often be efficiently evaluated in this setting. Examples of models that satisfy this factorisation structure are in fact quite common – and include Gaussian process regression models [Csató et al., 2000], some independent components analysis models [Højen-Sørensen et al., 2002], and linear regression models with Gaussian noise and factorising priors on the weights $\mathbf{w}$ [Titsias and Lázaro-Gredilla, 2012].

Approximation to $p(\mathbf{w})$

For problems where the site-projections depend on only a single element of the parameter vector such that $\prod_{n=1}^N \phi_n(\mathbf{w}^\mathsf{T}\mathbf{h}_n) = \prod_{d=1}^D \phi_d(w_d)$, the variational mean field approximation to the posterior is $q(\mathbf{w}) = \prod_d q(w_d)$ with the factors $q(w_d)$ defined as

$$q(w_d) = \frac{1}{Z_d}\phi_d(w_d)\mathcal{N}\left(w_d \,\middle|\, \frac{a_d}{\Lambda_{dd}}, \Lambda_{dd}^{-1}\right), \tag{3.4.2}$$

where the constants a_d are defined as expectations of the Gaussian potentials contribution to the KL bound taken with respect to the remaining variational factors $\{q(w_j)\}_{j\neq d}$, $\Lambda_{dd} := [\boldsymbol{\Sigma}^{-1}]_{dd}$ and Z_d is

the factor's normalisation constant – see Appendix A.7.1 for their precise forms.

Since the factorised approximation $q(\mathbf{w})$ in equation (3.4.2) is generally not of a simple analytic form, computing expectations with respect to $q(\mathbf{w})$ may be more computationally demanding than for Gaussian or delta approximations to $p(\mathbf{w})$.

Approximation to Z

For $q(\mathbf{w}) = \prod_d q(w_d)$, with each factor as defined as in equation (3.4.2), the mean field bound can be expressed as

$$\log Z \geq \mathcal{B}_{MF} = \sum_d H[q(w_d)] + \sum_d \langle \log \phi_d(w_d) \rangle_{q(w_d)} - \frac{1}{2}\left[\log\det(2\pi\boldsymbol{\Sigma}) + (\mathbf{m}-\boldsymbol{\mu})^{\mathsf{T}}\boldsymbol{\Sigma}^{-1}(\mathbf{m}-\boldsymbol{\mu}) + \mathbf{s}^{\mathsf{T}}\mathrm{diag}\left(\boldsymbol{\Sigma}^{-1}\right)\right], \quad (3.4.3)$$

where $[\mathbf{m}]_k := \langle w_k \rangle_{q(w_k)}$ and $[\mathbf{s}]_k := \left\langle (w_k - m_k)^2 \right\rangle_{q(w_k)}$.

Optimisation and complexity

Evaluating the MF bound, equation (3.4.3), and updating the variational factors requires the evaluation of D univariate expectations with respect to $q(w_d)$. Whether these expectations have simple analytic expressions depends on the specific form of the non-Gaussian potentials $\{\phi_n\}$. We note, however, that since the expectations are univariate with a Gaussian factor the domain for which they have significant mass will typically be easy to assess. Thus it is likely that the required expectations can be efficiently and accurately computed using univariate numerical integration routines. The mean field bound is generally cheap to compute – scaling $O\left(D^2\right)$ due to the inner product $\mathbf{m}^{\mathsf{T}}\boldsymbol{\Sigma}^{-1}\mathbf{m}$ assuming the precision matrix $\boldsymbol{\Sigma}^{-1}$ has been pre-computed. In general then, the mean field approximation requires just $O\left(D\right)$ memory and $O\left(D^2\right)$ time to evaluate and update the bound. Thus the MF method is generally one of the fastest global approximation methods considered in this chapter.

Qualities of approximation

The mean field method constructs a global approximation to the target since it optimises the KL divergence and so seeks to approximate $p(\mathbf{w})$ over its entire support. Mean field methods do not in general suffer from some of the pathologies of local methods such as the MAP and the Laplace approximation. Furthermore, since the factors are optimal subject only to the factorisation assumption we should expect the optimal approximation $q(\mathbf{w})$ to accurately capture the axis aligned marginals of the target density. However, ignoring all correlation in $p(\mathbf{w})$ is a restrictive assumption and renders this approximation unsuitable in many applications. For example, in Gaussian processes regression models and active learning procedures approximating target density covariance can be critical.

Since the mean field method is a $\mathrm{KL}(q(\mathbf{w})|p(\mathbf{w}))$ bound optimisation procedure, using its approximation $q(\mathbf{w})$ as the E-step density in an approximate EM optimisation procedure is guaranteed to increase a lower-bound on the likelihood.

3.5 Laplace approximation

The Laplace method approximates the target density by a multivariate Gaussian centred at its mode with covariance equal to the negative inverse Hessian of $\log p(\mathbf{w})$. It is thus equivalent to making a second order Taylor expansion of $\log p(\mathbf{w})$, with the expansion point the MAP estimate. Therefore, the Laplace approximation can be interpreted as an extension or improvement to the MAP approximation. As such it inherits some of the same qualities of the MAP approximate inference method.

The only restriction the Laplace approximation places on the site-projection potentials, since it is a Taylor expansion approximation, is that the second order derivatives exist at the mode of $p(\mathbf{w})$. Note that some models of significant utility do not satisfy this condition. For example, models with sparse Laplace potentials $\phi(w) \propto e^{-|w|}$ are often not differentiable at their modes [Seeger, 2008, Kuss, 2006].

Approximation to $p(\mathbf{w})$

The Taylor expansion to $\log p(\mathbf{w})$ implies an exponentiated quadratic approximation to $p(\mathbf{w})$. That is, the Laplace method approximates the target $p(\mathbf{w})$ by a multivariate Gaussian $q(\mathbf{w}) = \mathcal{N}(\mathbf{w}|\mathbf{m}, \mathbf{S})$ where

$$\begin{aligned} \mathbf{m} &:= \underset{\mathbf{w}}{\operatorname{argmax}} \ \log \mathcal{N}(\mathbf{w}|\boldsymbol{\mu}, \boldsymbol{\Sigma}) + \sum_{n=1}^{N} \log \phi_n(\mathbf{w}^\mathsf{T}\mathbf{h}_n), \\ \mathbf{S}^{-1} &:= \frac{\partial^2}{\partial \mathbf{w} \partial \mathbf{w}^\mathsf{T}} \log p(\mathbf{w}) \Big|_{\mathbf{w}=\mathbf{m}} = \boldsymbol{\Sigma}^{-1} + \mathbf{X}\boldsymbol{\Gamma}\mathbf{X}^\mathsf{T}, \end{aligned} \tag{3.5.1}$$

where $\boldsymbol{\Gamma}$ is an $N \times N$ diagonal matrix such that $\Gamma_{nn} = \psi_n''(\mathbf{m}^\mathsf{T}\mathbf{h}_n)$, $\psi_n := \log \phi_n$ and $\psi_n''(x) := \frac{\partial^2}{\partial x^2}\psi_n(x)$.

Approximation to Z

Substituting the second order Taylor approximation into the definition of the normalisation constant Z in equation (2.3.2), integrating and then taking the logarithm, the Laplace approximation provides the following approximation to $\log Z$

$$\log Z \approx \log Z_{Lap} = \log \det(2\pi\mathbf{S}) - \frac{1}{2}\left[\log \det(2\pi\boldsymbol{\Sigma}) + (\mathbf{m}-\boldsymbol{\mu})^\mathsf{T}\boldsymbol{\Sigma}^{-1}(\mathbf{m}-\boldsymbol{\mu})\right] + \sum_{n=1}^{N} \psi_n(\mathbf{m}^\mathsf{T}\mathbf{h}_n).$$

Optimisation and complexity

Computing the posterior mode is the MAP estimation problem considered in Section 3.3. Since the Laplace approximation requires that the target is twice continuously differentiable, optimisation is typically performed using a (approximate) second order gradient ascent procedure. For example, when the potentials are log-concave optimisation can typically be performed very efficiently using Newton's method. As we saw in Section 3.3 computing the MAP objective and its gradient scales $O(D(D+N))$.

Computing the Laplace approximation to the marginal likelihood requires computing the log determinant of the matrix $\mathbf{S}$, which is defined in equation (3.5.1). Importantly, since the value Z_{Lap} is not required during optimisation, this term only needs to be computed once. Due to the structure of the covariance, as explained in Section 2.1.1, we can compute this term in $O(ND\min\{N, D\})$ time.

Since cubic matrix operations only need to be performed once, the Laplace approximation is highly scalable. As we have discussed, the optimisation task is similar to that of the MAP approximation. In larger problems, when computing the full Hessian is infeasible, we can approximate it either by computing only a subset of its elements (for instance just its diagonal elements) or we can construct some low rank eigenvector decomposition of it. For the latter approach, approximations to its leading eigenvectors may be approximated, for example, using iterative Lanczos methods [Golub and Van Loan, 1996, Seeger, 2010].

Qualities of approximation

The Laplace method makes an essentially local approximation to the target. The Gaussian approximation to $p(\mathbf{w})$ is centred at the MAP estimate and so the Laplace approximation inherits some of the pathologies of that approximation. For example, if the mode is not representative of the target density, that is the mode has locally negligible mass, the Laplace approximation will be poor.

If the target is Gaussian, however, the Laplace approximation is exact. From the central limit theorem, we know in the limit of many data points, for a problem of fixed dimensionality, and certain other regularity conditions holding, the posterior will tend to a Gaussian centred at the posterior mode. Thus the Laplace approximation will become increasingly accurate in the limit of increasing data. Otherwise, in problems where D and N are the same order of magnitude the accuracy of the approximation will be governed by how Gaussian the target density is. Unimodality and log-concavity of the target are reasonable conditions under which we may expect the Laplace approximation to be effective.

Using Laplace approximations to $\{p(\mathbf{w}|\mathbf{v}, \boldsymbol{\theta})\}$ for the E-step of an approximate EM algorithm is not guaranteed to increase the likelihood or a lower-bound on it and can converge to a degenerate solution.

3.6 Gaussian expectation propagation approximation

Gaussian Expectation Propagation (G-EP) seeks to approximate the target density by sequentially matching moments between marginals of the variational Gaussian approximation and a density constructed from the variational Gaussian and an individual site potential [Minka, 2001a,b]. G-EP can be viewed as an iterative refinement of a one pass Gaussian density filtering approximation. Gaussian density filtering, and the equations necessary to implement it, is presented in Appendix A.2.5.

Approximation to $p(\mathbf{w})$

Gaussian EP approximates the target by a product of scaled Gaussian factors with the same factorisation structure as $p(\mathbf{w})$, so that

$$q(\mathbf{w}) := \frac{1}{Z}\mathcal{N}\left(\mathbf{w}|\boldsymbol{\mu}, \boldsymbol{\Sigma}\right) \prod_{n=1}^{N} \tilde{\phi}_n(\mathbf{w}^\mathsf{T}\mathbf{h}_n) = \mathcal{N}\left(\mathbf{w}|\mathbf{m}, \mathbf{S}\right), \tag{3.6.1}$$

where $\tilde{\phi}_n(\mathbf{w}^\mathsf{T}\mathbf{h}_n)$ are scaled Gaussian factors defined as

$$\tilde{\phi}_n(\mathbf{w}^\mathsf{T}\mathbf{h}_n) := \tilde{\gamma}_n e^{-\frac{1}{2\tilde{\sigma}_n^2}\left(\mathbf{w}^\mathsf{T}\mathbf{h}_n - \tilde{\mu}_n\right)^2}, \tag{3.6.2}$$

with variational parameters $\tilde{\gamma}_n, \tilde{\mu}_n$ and $\tilde{\sigma}_n^2$. Since exponentiated quadratics are closed under multiplication, $q(\mathbf{w})$ is Gaussian distributed also. We denote this global Gaussian approximation as $\mathcal{N}\left(\mathbf{w}|\mathbf{m}, \mathbf{S}\right)$.

Individually, each scaled Gaussian factor in equation (3.6.2) can be interpreted as a Gaussian approximation to the site-projection potential $\phi_n(\mathbf{w}^\mathsf{T}\mathbf{h}_n)$. The scaled Gaussian factors $\tilde{\phi}_n(\mathbf{w}^\mathsf{T}\mathbf{h}_n)$ in equation (3.6.2) are optimal for the univariate site-projection potentials $\phi_n(\mathbf{w}^\mathsf{T}\mathbf{h}_n)$ we consider here. For general inference problems G-EP site approximations would require full rank covariance and vector mean variational parameters.

The moments of the global Gaussian approximation $\mathbf{m}, \mathbf{S}$ are defined by the variational parameters $\left\{\tilde{\mu}_n, \tilde{\sigma}_n^2\right\}_{n=1}^N$ and the constants of the inference problem $\boldsymbol{\mu}, \boldsymbol{\Sigma}$ and $\mathbf{H}$, where $\mathbf{H} := [\mathbf{h}_1, ..., \mathbf{h}_N]$. Equations for $\mathbf{m}, \mathbf{S}$ can be obtained by equating first and second order terms in the exponents of equation (3.6.1).

Approximation to Z

Gaussian EP can provide an approximation to the normalisation constant Z by substituting the Gaussian approximate site approximations $\tilde{\phi}_n(\mathbf{w}^\mathsf{T}\mathbf{h}_n)$ in place of the intractable sites $\phi_n(\mathbf{w}^\mathsf{T}\mathbf{h}_n)$ and integrating, so that

$$Z \approx Z_{EP} := \int \mathcal{N}(\mathbf{w}|\boldsymbol{\mu}, \boldsymbol{\Sigma}) \prod_{n=1}^{N} \tilde{\phi}_n(\mathbf{w}^\mathsf{T}\mathbf{h}_n) d\mathbf{w}. \tag{3.6.3}$$

The integral above is tractable since it is an exponentiated quadratic. Completing the square in equation (3.6.3), integrating and taking its logarithms we arrive at

$$\log Z_{EP} = \frac{1}{2}\log\det(2\pi\mathbf{S}) - \frac{1}{2}\boldsymbol{\mu}^\mathsf{T}\boldsymbol{\Sigma}^{-1}\boldsymbol{\mu} - \tilde{\boldsymbol{\mu}}^\mathsf{T}\tilde{\boldsymbol{\Sigma}}\boldsymbol{\mu} + \frac{1}{2}\mathbf{m}^\mathsf{T}\mathbf{S}^{-1}\mathbf{m} - \frac{1}{2}\log\det(2\pi\boldsymbol{\Sigma}) + \sum_n \log\tilde{\gamma}_n, \tag{3.6.4}$$

where $\tilde{\boldsymbol{\mu}} := [\tilde{\mu}_1, ..., \tilde{\mu}_N]^\mathsf{T}$ and $\tilde{\boldsymbol{\Sigma}} = \operatorname{diag}\left([\tilde{\sigma}_1^2, ..., \tilde{\sigma}_N]^\mathsf{T}\right)$. Similarly to the Laplace approximation, computing the EP approximation to $\log Z$ in general scales $O(ND\min\{N, D\})$ due to the $\log\det(\mathbf{S})$ term.

Optimisation and complexity

EP can be viewed as an approximate method to optimise $\mathrm{KL}(p(\mathbf{w})|q(\mathbf{w}))$. Since the exact criterion is analytically intractable, because it requires computing expectations with respect to the intractable density $p(\mathbf{w})$, G-EP sequentially optimises a different, loosened objective. At each iteration of the G-EP optimisation procedure the following criterion is optimised with respect to the moments $\mathbf{m}$ and $\mathbf{S}$

$$\mathrm{KL}\left(\frac{q_{\setminus n}(\mathbf{w})\phi_n(\mathbf{w}^\mathsf{T}\mathbf{h}_n)}{Z_n}\middle\| q(\mathbf{w})\right), \tag{3.6.5}$$

where $q_{\setminus n}(\mathbf{w})$ denotes a Gaussian cavity potential defined by the product in equation (3.6.1) on omitting the factor $\tilde{\phi}_n$ so that $q_{\setminus n}(\mathbf{w}) = q(\mathbf{w})/\tilde{\phi}_n(\mathbf{w}^\mathsf{T}\mathbf{h}_n)$ and Z_n is the normalisation constant such that the product $q_{\setminus n}(\mathbf{w})\phi_n(\mathbf{w}^\mathsf{T}\mathbf{h}_n)$ is a probability density. The moments of the Gaussian cavity $q_{\setminus n}(\mathbf{w})$ can be obtained analytically using the results presented in Appendix A.2 – the mean and covariance of which are denoted $\mathbf{m}_{\setminus n}$ and $\mathbf{S}_{\setminus n}$ respectively.

For Gaussian $q(\mathbf{w})$, as shown in Section 3.2, equation (3.6.5) is minimised when the moments of $q(\mathbf{w}) = \mathcal{N}(\mathbf{w}|\mathbf{m}, \mathbf{S})$ match the moments of the 'tilted' density $q_{\setminus n}(\mathbf{w})\phi_n(\mathbf{w}^\mathsf{T}\mathbf{h}_n)/Z_n$. The moments

of $q_{\backslash n}(\mathbf{w})\phi_n(\mathbf{w}^\mathsf{T}\mathbf{h}_n)/Z_n$ can be computed using the Gaussian density filtering equations as described in Appendix A.2.5.

The G-EP algorithm initialises the Gaussian approximation's moments with the prior's: $\mathbf{m} \leftarrow \boldsymbol{\mu}$, $\mathbf{S} \leftarrow \boldsymbol{\Sigma}$. The algorithm then iteratively updates the approximating moments by cycling through each of the site potentials applying the following procedure until all $\tilde{\gamma}_n, \tilde{\mu}_n, \tilde{\sigma}_n^2$ have converged. To update the approximation for site n: first, calculate the Gaussian cavity $q_{\backslash n}(\mathbf{w}) = \mathcal{N}\left(\mathbf{w}|\mathbf{m}_{\backslash n}, \mathbf{S}_{\backslash n}\right)$; second, use the Gaussian density filtering equations to match moments between $q_{\backslash n}(\mathbf{w})\phi_n(\mathbf{w}^\mathsf{T}\mathbf{h}_n)/Z_n$ and $\mathcal{N}\left(\mathbf{w}|\mathbf{m}, \mathbf{S}\right)$ obtaining $\mathbf{m}^{new}, \mathbf{S}^{new}$; third, update the factor parameters $\tilde{\gamma}_n, \tilde{\mu}_n, \tilde{\sigma}_n^2$ so that $\mathcal{N}\left(\mathbf{w}|\mathbf{m}^{new}, \mathbf{S}^{new}\right) \propto q_{\backslash n}(\mathbf{w})\tilde{\phi}_n(\mathbf{w}^\mathsf{T}\mathbf{h}_n)$ and $Z_n = q_{\backslash n}(\mathbf{w})\tilde{\phi}_n(\mathbf{w}^\mathsf{T}\mathbf{h}_n)$.

The iterative G-EP fixed point procedure described above is not guaranteed to converge. Indeed, G-EP is frequently found to be unstable if the potentials $\{\phi_n\}$ are not log-concave [Seeger et al., 2007, Seeger, 2005]. Furthermore, if ϕ_n are log-concave but the Gaussian covariance matrix $\boldsymbol{\Sigma}$ is poorly conditioned, convergence issues can also occur, as is the case for the Gaussian process regression results presented in Section 5.1. G-EP places no restrictions on the potential functions regarding continuity or differentiability. Some G-EP convergence issues can be alleviated by adapting the local site fixed point update conditions so that the non-Gaussian potentials are taken to non-unity powers. This procedure, referred to as fractional or power G-EP, optimises a fundamentally different criterion to vanilla G-EP but is often observed to provide reasonable inferences and more robust convergence – see Minka [2004] and references therein for details.

An efficient implementation of Gaussian EP maintains the covariance matrix using the Cholesky decomposition of its inverse the precision matrix $\mathbf{S}^{-1} = \mathbf{P}^\mathsf{T}\mathbf{P}$ – see Appendix A.6.1. Using this factorisation, the computational bottleneck of a G-EP site update comes from computing a rank one update of the precision matrix, $\mathbf{P}^\mathsf{T}\mathbf{P} \leftarrow \mathbf{P}^\mathsf{T}\mathbf{P} + \nu\mathbf{h}_n\mathbf{h}_n^\mathsf{T}$, and solving a symmetric $D \times D$ linear system, where both of these computations scale $O\left(D^2\right)$. Thus a single G-EP iteration, where each of the non-Gaussian site potentials is updated once, scales $O\left(ND^2\right)$.

Provably convergent double loop extensions to G-EP have been developed – see Opper and Winther [2005] and references therein for details. Typically, these methods are slower than vanilla EP implementations. However, recent algorithmic developments have yielded significant speed-ups over vanilla EP whilst maintaining the convergence guarantees [Seeger and Nickisch, 2011a]. Importantly, however, these procedures still require the exact solution of rank D symmetric linear systems and thus scale $O\left(ND^2\right)$ in general.

G-EP and its provably convergent extensions have been shown to be unstable/infeasible if the precision matrix $\mathbf{S}^{-1}$ is not computed exactly [Seeger and Nickisch, 2011a]. Therefore, in larger problems where $O\left(D^2\right)$ memory and $O\left(ND^2\right)$ time requirements are not practical G-EP approximate inference is infeasible. Thus G-EP methods seem inherently unscalable for general, densely-connected latent linear models.

Qualities of approximation

Whilst G-EP can suffer from convergence issues, in log-concave models when convergence issues do not arise, it is often reported to be one of the most accurate Gaussian deterministic approximate inference methods. Nickisch and Rasmussen [2008] applied each of the approximate inference methods considered here to a Gaussian process binary logistic regression model. The logistic sigmoidal conditional likelihood is log-concave and G-EP was numerically stable. Comparing G-EP versus Laplace, local variational bounding, and Gaussian KL bounding, G-EP often achieved the most accurate inferences regarding the approximation of both Z and $p(\mathbf{w})$.

Since the G-EP procedure does not optimise a lower-bound on Z using its approximation to $p(\mathbf{w})$ for the E-step of an approximate EM or gradient ascent maximum likelihood optimisation procedure is not guaranteed to increase the likelihood or a lower-bound on it. However, if the G-EP approximate inference procedure converges the approximation can often perform well – for example see Kim and Ghahramani [2006], Nickisch and Rasmussen [2008].

3.7 Gaussian Kullback-Leibler bounding

Gaussian Kullback-Leibler (G-KL) approximate inference seeks to approximate the target $p(\mathbf{w})$ by minimising the KL divergence $\mathrm{KL}(q(\mathbf{w})|p(\mathbf{w}))$ with the variational density $q(\mathbf{w})$ constrained to be a multivariate Gaussian. G-KL approximate inference was originally presented by [Hinton and Van Camp, 1993] for factorising Gaussian approximations and Barber and Bishop [1998a] for full covariance Gaussian approximations. Until recently (see for example Opper and Archambeau [2009], Honkela and Valpola [2005]) G-KL methods have received little attention by the research community in comparison to the other deterministic methods considered in this chapter. Principally this is due to the perceived unfavourable computational demands of the approximation. G-KL approximate inference and the developments we have made regarding this procedure are the focus of Chapter 4. In this section we present the G-KL method as proposed in work prior to our contributions.

G-KL approximate inference methods are widely applicable placing few functional restrictions on the site potentials $\{\phi_n\}$ to which it can be applied. All that is required, for the KL divergence to be well defined, is that each site potential has unbounded support.

Approximation to $p(\mathbf{w})$

G-KL approximate inference obtains the 'best' Gaussian approximation $q^*(\mathbf{w}) := \mathcal{N}(\mathbf{w}|\mathbf{m}^*, \mathbf{S}^*)$ to the target $p(\mathbf{w})$ by minimising the KL divergence $\mathrm{KL}(q(\mathbf{w})|p(\mathbf{w}))$ with respect to the moments $\mathbf{m}, \mathbf{S}$, so that

$$\begin{aligned}\mathbf{m}^*, \mathbf{S}^* &:= \operatorname*{argmin}_{\mathbf{m},\mathbf{S}} \mathrm{KL}(q(\mathbf{w})|p(\mathbf{w})) \\ &= \operatorname*{argmax}_{\mathbf{m},\mathbf{S}} H\left[q(\mathbf{w})\right] + \left\langle \log \mathcal{N}\left(\mathbf{w}|\boldsymbol{\mu}, \boldsymbol{\Sigma}\right)\right\rangle + \sum_{n=1}^{N} \left\langle \log \phi_n(\mathbf{w}^\mathsf{T}\mathbf{h}_n)\right\rangle, \end{aligned} \tag{3.7.1}$$

where the expectations in equation (3.7.1) are taken with respect to the variational Gaussian $q(\mathbf{w}) = \mathcal{N}(\mathbf{w}|\mathbf{m}, \mathbf{S})$.

Approximation to Z

As a $\mathrm{KL}(q(\mathbf{w})|p(\mathbf{w}))$ approximate inference method, see Section 3.2, the G-KL method provides the following lower-bound on the normalisation constant

$$\log Z \geq \mathcal{B}_{G\text{-}KL} := \underbrace{\frac{1}{2}\log\det\left(2\pi e\mathbf{S}\right)}_{\text{entropy}} + \underbrace{\sum_{n=1}^{N}\left\langle\log\phi_n(m_n+zs_n)\right\rangle_{\mathcal{N}(z|0,1)}}_{\text{site-projection potentials}} - \underbrace{\frac{1}{2}\left[\log\det\left(2\pi\boldsymbol{\Sigma}\right)+\left(\mathbf{m}-\boldsymbol{\mu}\right)^{\mathsf{T}}\boldsymbol{\Sigma}^{-1}\left(\mathbf{m}-\boldsymbol{\mu}\right)+\mathrm{trace}\left(\boldsymbol{\Sigma}^{-1}\mathbf{S}\right)\right]}_{\text{Gaussian potential}}, \quad (3.7.2)$$

where we have used the fact that the expectation $\left\langle\log\phi_n(\mathbf{w}^{\mathsf{T}}\mathbf{h}_n)\right\rangle_{q(\mathbf{w})} = \left\langle\log\phi(m_n+zs_n)\right\rangle_{\mathcal{N}(z|0,1)}$ where $m_n := \mathbf{m}^{\mathsf{T}}\mathbf{h}_n$ and $s_n^2 := \mathbf{h}_n^{\mathsf{T}}\mathbf{S}\mathbf{h}_n$ – this result, due to Barber and Bishop [1998a], is presented in Appendix A.2.4. For some potential functions the univariate expectation $\left\langle\log\phi_n(m_n+zs_n)\right\rangle$ will have a simple analytic form. For example, this is the case for Laplace potentials where $\phi(x)\propto e^{-|x|}$. When this expectation cannot be analytically derived we note that since it is a univariate Gaussian expectation it can typically be computed efficiently using numerical methods.

Optimisation and complexity

G-KL approximate inference proceeds by solving the optimisation problem posed in equation (3.7.1). Typically this is implemented by performing gradient ascent in the moments $\mathbf{m},\mathbf{S}$ using approximate second order methods.

Specifying the G-KL mean and covariance requires $\frac{1}{2}D(D+3)$ parameters which can be a much larger optimisation task than that required by local variational bounding methods with N variational parameters, the Laplace approximation with D variational parameters or the G-EP approximation with $2N$ variational parameters. The size of the G-KL optimisation problem is proffered as the reason for why other variational methods have been favoured over G-KL methods [Opper and Archambeau, 2009].

Seeger [1999] showed that the Gaussian covariance $\mathbf{S}$ can be parameterised using N variational parameters by noting that if we differentiate equation (3.7.2) with respect to $\mathbf{S}$ and equate the derivative to zero, the optimal covariance has the structure

$$\mathbf{S}^{-1} = \boldsymbol{\Sigma}^{-1} + \mathbf{H}\boldsymbol{\Gamma}\mathbf{H}^{\mathsf{T}}, \quad (3.7.3)$$

where $\mathbf{H}=[\mathbf{h}_1,...,\mathbf{h}_N]$ and $\boldsymbol{\Gamma}\in\mathbb{R}^{N\times N}$ is diagonal. Thus we can parameterise the G-KL covariance using just the N diagonal elements of $\boldsymbol{\Gamma}$ as parameters. In many modelling scenarios $N \ll \frac{1}{2}D(D+1)$ and this parameterisation will result in a significant reduction in the size of the optimisation space, and thus, hopefully, also a commensurate reduction in the complexity of the optimisation problem. However, using this parameterisation of G-KL covariance does not significantly reduce the cost of the matrix computations that are required to evaluate the G-KL bound and its derivatives. Using the matrix inversion and matrix determinant lemmas, $\mathbf{S}$ and $\log\det\left(\mathbf{S}\right)$ can be computed in $O\left(ND\min\left\{D,N\right\}\right)$ time, whereas directly evaluating these terms using the unstructured covariance matrix scales $O\left(D^3\right)$.

Since the G-KL method is a lower-bound optimisation procedure it is typically very numerically stable. G-KL approximate inference does not require that the targets be log-concave like the G-EP ap-

proximation, differentiable like the Laplace approximation or super-Gaussian like local lower-bounding methods. Even if the target density $p(\mathbf{w})$ is multimodal the G-KL solution will converge to cover one of these modes.

Regardless of which parameterisation is used for the G-KL covariance $\mathbf{S}$, G-KL approximate inference, as presented here, is not a scalable optimisation procedure. Evaluating the bound and computing its gradient requires multiple cubic matrix operations (for example the $\log\det(\mathbf{S})$ term and its derivatives) which need to be computed many times by any gradient ascent procedure. Furthermore, the objective, parameterised either directly in $\mathbf{S}$ or with respect to the diagonal elements of $\boldsymbol{\Gamma}$ using equation (3.7.3), is neither concave nor convex. Techniques to overcome these computational limitations are presented in Chapter 4, and are one of the core contributions of this thesis.

Qualities of approximation

As we saw in Section 3.2 optimising the $\mathrm{KL}(q(\mathbf{w})|p(\mathbf{w}))$ objective tends to result in approximations that are mode seeking or avoid making false positive predictions. However, if our posterior is unimodal, or is well approximated by a single mode (as is the case, for instance, in mixture models where index permutations describe equivalent densities), the $\mathrm{KL}(q(\mathbf{w})|p(\mathbf{w}))$ objective for Gaussian $q(\mathbf{w})$ can be expected to perform well. Nickisch and Rasmussen [2008] presented a thorough comparison of deterministic approximate inference methods for Gaussian process logistic regression models. Their results showed that G-KL approximate inference (alongside the G-EP method) was amongst the most accurate methods considered.

Opper and Archambeau [2009] showed that the G-KL approximation can be interpreted as an 'averaged' Laplace approximation. Laplace approximate moments, $\mathbf{m}$ and $\mathbf{S}$, are defined as satisfying the following critical point conditions:

$$\mathbf{0} = \frac{\partial}{\partial \mathbf{w}} \log \mathcal{N}(\mathbf{w}|\boldsymbol{\mu}, \boldsymbol{\Sigma}) \prod_n \phi_n(\mathbf{w}^\mathsf{T}\mathbf{h}_n),$$

$$\mathbf{S}^{-1} := -\frac{\partial^2}{\partial \mathbf{w}\mathbf{w}^\mathsf{T}} \log \mathcal{N}(\mathbf{w}|\boldsymbol{\mu}, \boldsymbol{\Sigma}) \prod_n \phi_n(\mathbf{w}^\mathsf{T}\mathbf{h}_n)\Bigg|_{\mathbf{w}=\mathbf{w}_{MAP}}.$$

Opper and Archambeau [2009] show that the optimal G-KL moments satisfy the following implicit equations:

$$\mathbf{0} = \left\langle \frac{\partial}{\partial \mathbf{w}} \log \mathcal{N}(\mathbf{w}|\boldsymbol{\mu}, \boldsymbol{\Sigma}) \prod_n \phi_n(\mathbf{w}^\mathsf{T}\mathbf{h}_n) \right\rangle,$$

$$\mathbf{S}^{-1} = -\left\langle \frac{\partial^2}{\partial \mathbf{w}\mathbf{w}^\mathsf{T}} \log \mathcal{N}(\mathbf{w}|\boldsymbol{\mu}, \boldsymbol{\Sigma}) \prod_n \phi_n(\mathbf{w}^\mathsf{T}\mathbf{h}_n) \right\rangle,$$

where the expectation $\langle\cdot\rangle$ is taken with respect to the G-KL distribution $q(\mathbf{w})$. This result provides an interpretation of the G-KL approximation as an 'averaged' Laplace approximation. Laplace approximations can be inaccurate since the inverse Hessian of $\log p(\mathbf{w})$ at its mode is a poor estimator of target density covariance. This discrepancy is due to $\log p(\mathbf{w})$ not being a quadratic thus a point estimate of its curvature cannot capture the covariance of the density $p(\mathbf{w})$. We might hope then that the G-KL approximation, which effectively averages the Laplace approximation over the support of $q(\mathbf{w})$ will be better at capturing target density covariance.

Since the G-KL procedure is a $\mathrm{KL}(q(\mathbf{w})|p(\mathbf{w}))$ lower-bounding method, using the Gaussian approximation $q(\mathbf{w})$ in the E-step in an approximate EM procedure is guaranteed to increase a lower-bound on the likelihood.

3.8 Local variational bounding

Local variational bounding (LVB) procedures lower-bound Z by replacing each non-Gaussian potential $\phi_n(\mathbf{w}^\mathsf{T}\mathbf{h}_n)$ in the integrand of Z with a function that lower-bounds it and that renders the integral as a whole analytically tractable. Tractability is obtained by utilising exponentiated quadratic lower-bounds for each site potential $\phi_n(\mathbf{w}^\mathsf{T}\mathbf{h}_n)$. Local variational bounding methods have received much attention from the research community over the years and have been employed for a wide variety of latent linear models – see for example Jaakkola and Jordan [1997], Saul et al. [1996], Seeger and Nickisch [2010], Gibbs and MacKay [2000], Girolami [2001].

The lower-bound for each non-Gaussian potential $\phi_n(\mathbf{w}^\mathsf{T}\mathbf{h}_n)$ is parameterised by a single variational parameter which we denote as ξ_n. See Figure 3.4 for a depiction of a logistic sigmoid and a Laplace potential, $\phi(x)$, with tight exponentiated quadratic lower-bounds evaluated at a particular operating point $x = x^*$. Since exponentiated quadratics are closed under multiplication, one may bound the product of site potentials by an exponentiated quadratic also so that

$$\prod_n \phi_n(\mathbf{w}^\mathsf{T}\mathbf{h}_n) \geq c(\boldsymbol{\xi}) e^{-\frac{1}{2}\mathbf{w}^\mathsf{T}\mathbf{F}(\boldsymbol{\xi})\mathbf{w} + \mathbf{w}^\mathsf{T}\mathbf{f}(\boldsymbol{\xi})}, \tag{3.8.1}$$

where the matrix $\mathbf{F}(\boldsymbol{\xi})$ takes the form of an outer product matrix with the 'data' vectors $\mathbf{H} := [\mathbf{h}_1, ..., \mathbf{h}_N]$ so that $\mathbf{F}(\boldsymbol{\xi}) = \mathbf{H}\boldsymbol{\Lambda}(\boldsymbol{\xi})\mathbf{H}^\mathsf{T}$ and $\boldsymbol{\Lambda}(\boldsymbol{\xi})$ is a $N \times N$ diagonal matrix with elements defined by each of the site potential bounds. Similarly, the vector $\mathbf{f}(\boldsymbol{\xi})$ and scalar $c(\boldsymbol{\xi})$ depend on the site bounds and the 'data' $\mathbf{H}$. The $\boldsymbol{\xi}$ vector is of length N containing each of the variational parameters ξ_n. For any setting of $\mathbf{w}$ there exists a setting of $\boldsymbol{\xi}$ for which the bound is tight.

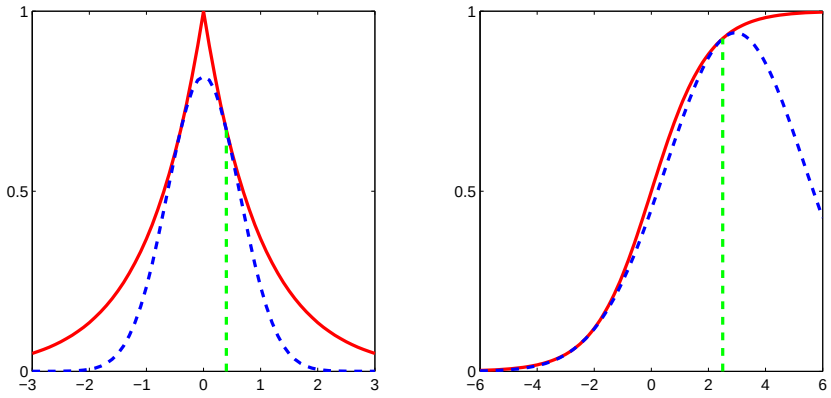
LVB procedures are applicable provided tight exponentiated quadratic lower-bounds to the site-projection potentials $\{\phi_n\}_{n=1}^N$ exist. Palmer et al. [2006] showed that such bounds exist provided the site potentials are super-Gaussian. Where they define a function $f(x)$ as super-Gaussian if $\exists b \in \mathbb{R}$ s.t. for $g(x) := \log f(x) - bx$ is even, convex and decreasing as a function of $y = x^2$. A number of potential functions of significant practical utility are super-Gaussian. Examples include: the logistic sigmoid $\phi(x) = (1 + \exp(-x))^{-1}$, the Laplace density $\phi(x) \propto \exp(-|x|)$ and the Student's t density. However, we note that deriving a new bound for each site potential we may want to use can be a non trivial task.

Approximation to $p(\mathbf{w})$

Substituting the site potential lower-bounds into the definition of $p(\mathbf{w})$ we can collect first and second order terms in the exponent to derive the moments of the Gaussian approximation to the target. Doing so gives $q(\mathbf{w}) = \mathcal{N}\left(\mathbf{w}|\mathbf{A}^{-1}\mathbf{b}, \mathbf{A}^{-1}\right)$ where

$$\mathbf{A} := \boldsymbol{\Sigma}^{-1} + \mathbf{F}(\boldsymbol{\xi}), \quad \mathbf{b} := \boldsymbol{\Sigma}^{-1}\boldsymbol{\mu} + \mathbf{f}(\boldsymbol{\xi}), \tag{3.8.2}$$

and so both $\mathbf{A}$ and $\mathbf{b}$ are functions of $\boldsymbol{\xi}$.



(a) (b)

Figure 3.4: Exponentiated quadratic lower-bounds for two super-Gaussian potential functions: (a) Laplace potential and lower-bound with operating point at 0.5; (b) Logistic sigmoid potential and lower-bound with operating point at 2.5.

Approximation to Z

Substituting equation (3.8.1) into the definition of Z in equation (2.3.2) we obtain the LVB bound on Z

$$\begin{aligned} Z &= \int \mathcal{N}\left(\mathbf{w}|\boldsymbol{\mu}, \boldsymbol{\Sigma}\right) \prod_{n=1}^{N} \phi_n(\mathbf{w}^\mathsf{T}\mathbf{h}_n) d\mathbf{w} \\ &\geq \int \mathcal{N}\left(\mathbf{w}|\boldsymbol{\mu}, \boldsymbol{\Sigma}\right) c(\boldsymbol{\xi}) e^{-\frac{1}{2}\mathbf{w}^\mathsf{T}\mathbf{F}(\boldsymbol{\xi})\mathbf{w} + \mathbf{w}^\mathsf{T}\mathbf{f}(\boldsymbol{\xi})} d\mathbf{w} \\ &= c(\boldsymbol{\xi}) \frac{e^{-\frac{1}{2}\boldsymbol{\mu}^\mathsf{T}\boldsymbol{\Sigma}^{-1}\boldsymbol{\mu}}}{\sqrt{\det\left(2\pi\boldsymbol{\Sigma}\right)}} \int e^{-\frac{1}{2}\mathbf{w}^\mathsf{T}\mathbf{A}\mathbf{w} + \mathbf{w}^\mathsf{T}\mathbf{b}} d\mathbf{w}. \end{aligned} \tag{3.8.3}$$

LVB methods obtain the tightest lower-bound on Z by optimising equation (3.8.3) with respect to the variational parameters $\boldsymbol{\xi}$. Optimisation can be implemented using either a gradient ascent or an EM procedure where $\boldsymbol{\xi}$ are treated as hyperparameters [Palmer et al., 2006].

Completing the square in equation (3.8.3), integrating and taking its logarithm, we have $\log Z \geq \mathcal{B}_{LVB}(\boldsymbol{\xi})$, where

$$\mathcal{B}_{LVB}(\boldsymbol{\xi}) = \log c(\boldsymbol{\xi}) - \frac{1}{2}\boldsymbol{\mu}^\mathsf{T}\boldsymbol{\Sigma}^{-1}\boldsymbol{\mu} + \frac{1}{2}\mathbf{b}^\mathsf{T}\mathbf{A}^{-1}\mathbf{b} - \frac{1}{2}\log\det\left(2\pi\boldsymbol{\Sigma}\right) - \frac{1}{2}\log\det\left(2\pi\mathbf{A}\right). \tag{3.8.4}$$

Optimisation and complexity

Evaluating $\mathcal{B}_{LVB}(\boldsymbol{\xi})$ requires computing $\log\det\left(\mathbf{A}\right)$ – typically the most computationally expensive term in equation (3.8.4). Assuming $\boldsymbol{\Sigma}^{-1}$ can be computed efficiently, $\mathbf{A}$ and its log determinant can typically be computed using the matrix inversion lemma and so scales $O\left(ND\min\left\{D, N\right\}\right)$.

Optimising the bound, using either expectation maximisation or gradient based methods, requires solving N linear symmetric $D \times D$ systems. Efficient exact implementations of this method maintain the covariance using its Cholesky factorisation and perform efficient rank one Cholesky updates [Seeger, 2007]. Doing so, each round of updates scales $O\left(ND^2\right)$.

Recently scalable approximate solvers for local variational bounding procedures have been developed – see Seeger and Nickisch [2011b] for a review. These methods make use of a number of algorithmic relaxations to reduce the computational burden of local bound optimisation. First, double loop algorithms are employed that reduce the number of times that $\log\det(\mathbf{A})$ and its derivatives need to be computed. Nickisch and Seeger [2009] also proved that for log-concave site potentials $\{\phi_n\}$ the LVB objective was a convex optimisation problem and hence global convergence rates could be expected to be rapid.

Second, these algorithms use approximate methods to evaluate the marginal variances that are required to drive local variational bound optimisation. Marginal variances are approximated either by constructing low rank factorisations of $\mathbf{A}$ using iterative Lanczos methods or by perturb and MAP sampling methods [Papandreou and Yuille, 2010, Seeger, 2010, Ko and Seeger, 2012]. Both of these approximations can greatly increase the speed of inference and the size of problems to which local procedures can be applied. Unfortunately, these relaxations are not without consequence regarding the quality of approximate inference. For example, the $\log\det(\mathbf{A})$ term is no longer exactly computed and a lower-bound on $\log Z$ is no longer maintained – only an estimate of $\log Z$ is provided. Using the Lanczos approximation marginal variances are often found to be strongly underestimated and bound values strongly overestimated. Whilst the scaling properties are, in general, problem and user dependent, roughly speaking, these relaxations reduce the computational complexity to scaling $O\left(KD^2\right)$ where K is the rank of the approximate covariance factorisation.

Qualities of approximation

One of the core advantages of local variational bounding methods is that they provide a principled lower-bound on Z. The lower-bound can be used as surrogate for the likelihood and we can derive convergent approximate EM parameter estimation methods.

Since LVB, G-KL and mean field methods all provide a lower-bound on Z, a natural question to ask is which of these is the tightest. The lower-bound to $\log Z$ presented in equation (3.8.4) is derived from a fundamentally different criterion from the mean field and G-KL bounds. Mean field and G-KL bounds are derived from the KL divergence $\mathrm{KL}(q(\mathbf{w})|p(\mathbf{w}))$. Local variational bounds are obtained by lower-bounding each non-Gaussian site-projection potential, integrating and optimising that analytic expression with respect to the parameters of the site bounds. Since we can substitute the LVB Gaussian approximation $q(\mathbf{w})$ into the KL divergence we can evaluate the KL bound using the LVB Gaussian. This observation has lead previous authors to suggest that when using local variational bounding methods, both bounds should be computed and the greater of the two should be used [Nickisch and Rasmussen, 2008]. Empirical evidence suggests that the LVB bound is poorer than the unconstrained G-KL bound. In Section 4.4.1 we address this issue specifically and show that in fact the G-KL bound is guaranteed to be stronger than that provided by LVB methods.

LVB methods provide a global Gaussian approximation to the target density $p(\mathbf{w})$ and as such can capture its correlational structure. Furthermore, since LVB methods are convex optimisation problems LVB methods should converge rapidly to their globally optimal parameter setting. LVB methods have

been reported to underestimate target density variance and the bound on the normalisation constant versus the G-KL and G-EP approximations [Nickisch and Rasmussen, 2008].

3.9 Comparisons

Each of the approximate inference methods considered above have their own relative strengths and weaknesses. Below we briefly discuss and compare these properties in relation to the desiderata laid out in Section 2.3.

Efficiency

Computationally, the MAP and the Laplace approximations are the fastest and most scalable methods considered. For some problems, where there is sufficient data relative to the dimensionality of parameter space, the MAP approximation, which ignores all uncertainty in the target, may be reasonable. If an estimate of the covariance of the target is required then the Laplace approximation can be used at moderate additional expense. The Laplace approximation is highly scalable in so much as approximations to the Hessian are easy to construct. However, the MAP and the Laplace methods are essentially local approximation methods that ignore measure in parameter space, and consequently can result in very poor approximations. Of the global approximation methods considered (mean field, G-EP, G-KL, and LVB), mean field and LVB methods are the most efficient and scalable techniques. However, mean field bounding methods, whilst scalable, are often not practical due to the requirement that the site potentials factorise such that $\prod_n \phi_n(\mathbf{w}^\mathsf{T}\mathbf{h}_n) = \prod_d \phi_d(w_d)$ and since a factorised approximation may be inadequate. LVB methods are the most efficient and scalable global approximate inference methods considered that do not place restrictive factorisation assumptions on the approximating density $q(\mathbf{w})$. Additionally, LVB methods are the only global approximate inference method (prior to our contribution) to result in a convex optimisation problem. Whilst G-EP and G-KL can be the most accurate methods considered here, they are currently not scalable.

Accuracy

With regards to accuracy of inference, the G-EP and the G-KL global approximation methods, for log-concave site potentials, have been reported to be the most accurate. LVB methods have been reported to underestimate both target density covariance and its normalisation constant. Mean field methods make strong factorisation assumptions about the target density approximation, which if unjustified, can result in poor approximations. As discussed, the local approximations made by the MAP and Laplace methods can, in some settings, result in extremely inaccurate approximate inferences.

With regards to performing parameter estimation and model selection, G-KL, mean field and LVB methods all provide a principled lower-bound on Z and so, using their respective criterions as a surrogate for the likelihood, numerically robust (hyper) parameter estimation and model selection procedures can be derived. Since the Laplace and G-EP methods are approximation methods, numerical issues can arise if we use these techniques to drive parameter estimation procedures. The LVB bound on Z is often reported to be weaker than the G-KL bound.

Laplace, G-EP, G-KL, and LVB methods all provide a Gaussian approximation to the target density

$p(\mathbf{w})$. Multivariate Gaussian densities have many desirable analytic and computational properties: simple analytic forms for conditionals and marginals can be derived, first and seconder order moments can be immediately accessed, the expectations of many functions can often be efficiently computed. The accuracy of a Gaussian approximation depends on the target density and the approximate inference routine used to find it. Independent of the approximation method used, Gaussian densities can easily capture and represent the correlation in the target density whereas mean field methods and MAP approximations cannot.

Generality

We saw that each of the approximate inference methods considered placed different constraints on the non-Gaussian potential functions to which they could be applied. The Laplace approximation requires that the site potentials are twice continuously differentiable. LVB methods require that the site potentials are super-Gaussian. G-EP methods can encounter convergence issues if the sites are not log-concave. The MAP approximation is highly inaccurate if the target density has non negligible variance. Tractability for mean field method requires that the site-projection potentials depend on only a single element of the parameter vector $\mathbf{w}$. G-KL bounding methods require that the support of the site potentials is unbounded.

3.10 Extensions

The deterministic approximate inference methods presented above are, in practice, the most commonly used methods in the latent linear model class. However, each of these techniques impose quite restrictive structures on the form of the approximating density: The MAP approximation uses a delta function density, the mean field method uses a fully factorising approximating density and the remaining methods use a multivariate Gaussian density. In many settings, the target may be far from being well modeled by such approximations. To that end we now briefly review some methods proposed in the literature to increase the flexibility of the class of approximating densities.

Mixture mean field

Mixture models, where a density is defined as a finite sum of simpler densities, are a commonly used technique to construct more flexible distributions. It is natural then to consider extending the KL bound optimisation techniques, such as the mean field and the G-KL methods, to optimise for $q(\mathbf{w})$ a mixture. For a variational density with K mixture components, the energy term of the KL bound is tractable and simply requires the evaluation of K times as many expectations as for standard KL methods. Computing the entropy term is less straightforward. The entropy of a mixture, $q(\mathbf{w}) := \sum_k \pi_k q_k(\mathbf{w})$ where $\sum_k \pi_k = 1$, is defined by the K integrals

$$H[q(\mathbf{w})] = -\sum_{k=1}^{K} \pi_k \int q_k(\mathbf{w}) \log \left(\sum_k \pi_k q_k(\mathbf{w}) \right) d\mathbf{w}. \tag{3.10.1}$$

The integrals presented in equation (3.10.1) are typically intractable due to the log sum structure. Bishop et al. [1998] propose to employ Jensen's bound, $\langle f(x) \rangle \geq f(\langle x \rangle)$ when $f(x)$ is convex, to take the negative log outside the expectation, in doing so one obtains a lower-bound on the entropy of the mixture.

The bounded entropy is then substituted into the KL bound to obtain a weakened, but tractable, lower-bound to $\log Z$. Accordingly, the class of approximating distributions $q(\mathbf{w})$ can be expanded to include mixture densities and so increase the accuracy of the mean field approximation. However, this procure has a two principal disadvantages: bounding the entropy further weakens the bound on Z, and evaluating the entropy's bound requires us to evaluate $O\left(K^2\right)$, D-dimensional expectations with respect to the mixture components $q_k(\mathbf{w})$ and so may not be that scalable or efficient.

When the expectation of the energy term cannot be computed exactly, people have proposed approximately computing their expectation. For example, Gershman et al. [2012] propose using a Gaussian mixture approximation and replace the site potentials with their first or second order Taylor expansions, thus a lower-bound on the normalisation constant is lost.

Split variational inference

A related approach to the mixture mean field method discussed above is the split variational inference technique [Bouchard and Zoeter, 2009]. Split variational inference methods develop the intuition that if we could partition the integral into a collection of smaller easier to approximate sub-integrals the accuracy of the approximation could be improved. To do this the authors consider a soft partition of the integral domain using binning functions such that $\sum_k b_k(\mathbf{w}) \equiv 1$ for all $\mathbf{w} \in \mathcal{W}$. Defining the target density $p(\mathbf{w}) = f(\mathbf{w})/Z$ we can see that

$$Z = \int f(\mathbf{w}) d\mathbf{w} = \sum_k \int b_k(\mathbf{w}) f(\mathbf{w}) d\mathbf{w} =: \sum_k Z_k.$$

Each sub-integral Z_k can then be lower-bounded by standard G-KL, LVB or factorising mean field methods as presented in the previous sections. The sum of the individual lower-bounds on Z_k then provides a global lower-bound on Z.

The authors propose a double loop optimisation procedure to perform split variational inference. In the outer loop, the global bound is optimised with respect to the binning functions that define the soft partitioning of the domain. In the inner loop each of the partitioned lower-bounds on Z_k are optimised. If the binning functions are taken to be softmax factors the split mean field method is equivalent to mixture mean field approach. Similarly to the mixture mean field method this method increases the accuracy of the approximation. However split variational inference can be quite computationally demanding since the inner loop of the optimisation procedure requires K lower-bound optimisation problems of D-dimensional integrals to be solved.

Skew-normal variational inference

Gaussian KL approximate inference minimises the $\mathrm{KL}(q(\mathbf{w})|p(\mathbf{w}))$ objective for $q(\mathbf{w})$ a multivariate Gaussian. This approach is practical since both the energy and entropy terms of the KL bound and their derivatives can be computed. Unfortunately, few other multivariate parametric densities are known for which these properties hold. One exception is the skew-normal density [Ormerod, 2011]. The skew normal variational approximate density $q(\mathbf{w})$ is defined as

$$q(\mathbf{w}|\mathbf{m}, \mathbf{S}, \mathbf{d}) := 2\mathcal{N}(\mathbf{w}|\mathbf{m}, \mathbf{S})\, \Phi(\mathbf{d}^{\mathsf{T}}(\mathbf{w} - \mathbf{m}))$$

where $\Phi(x) := \int_{-\infty}^{x} \mathcal{N}(t|0,1)$ is the standard normal cumulative density function. Thus the skew-normal density described in the equation above is a one dimensional distortion of a multivariate normal in the direction of the parameter vector $\mathbf{d}$. This density is more flexible than the multivariate Gaussian (if $\mathbf{d} = \mathbf{0}$ we recover the multivariate Gaussian) and so it can achieve more accurate inferences. However it is still a relatively constrained density and so the improvement it can achieve over standard Gaussian KL is relatively limited.

3.11 Summary

In this chapter we have reviewed some of the most popular approaches to deterministic approximate inference in the latent linear model class.

We have seen how the multivariate Gaussian density can be used as a flexible yet compact approximation to the target density and how this approximation, with its convenient computational properties, can render down stream computations such as expectations tractable. The Gaussian EP, the Laplace, the Gaussian KL and local variational bounding approximations all return a multivariate Gaussian approximation to the target. Each method placed different constraints on the potential functions to which it could be applied, had different computational complexity and scalability properties and resulted in different approximate inferences. Of each of these methods the G-KL approach has received by far the least attention by the research community. Principally this is due to the perceived computational complexity of G-KL bound evaluation and optimisation procedures. In Chapter 4 we consider G-KL approximate inferences in some depth, and present a range of methods and results that show that in fact this method is both efficient and scalable.

This chapter also drew attention to some of the limitations of the commonly used deterministic approximate inference methods. Whilst often convenient to implement and computationally fast, the approximate methods we present in the first part of this chapter can be inadequate if our target density is far from Gaussian distributed. In Section 3.10 we presented a few extensions that have been proposed in the literature to enrich the class of variational approximating distributions. In Chapter 6 we develop on these methods, providing a new method to evaluate and optimise the KL bound over a broad class of multivariate approximating densities.

Chapter 4

Gaussian KL approximate inference

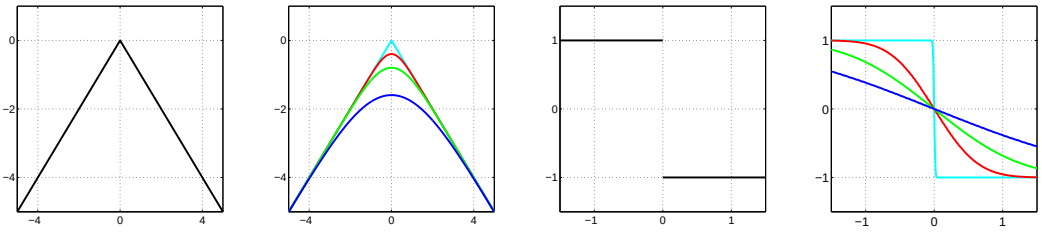
In this chapter we provide a number of novel contributions regarding the application of Gaussian Kullback-Leibler (G-KL) approximate inference methods to latent linear models. In Section 4.2 we address G-KL bound optimisation. We provide conditions on the potential functions $\{\phi_n\}_{n=1}^N$ for which the G-KL bound is smooth and concave. Thus we provide conditions for which optimisation using Newton's method will exhibit quadratic convergence rates and using quasi-Newton methods super-linear convergence rates. In Section 4.3 we discuss the complexity of G-KL bound and gradient computations required to perform approximate inference. To make G-KL approximate inference scalable we present constrained parameterisations of covariance. In Section 4.4 we compare G-KL approximate inference to other Gaussian approximate inference methods. We prove that the G-KL lower-bound is tighter than the bound offered by local lower-bounding methods. We also discuss and compare computational scaling properties and model applicability issues.

4.1 Introduction

As introduced in Section 3.7, G-KL approximate inference proceeds by fitting the variational Gaussian, $q(\mathbf{w}) = \mathcal{N}(\mathbf{w}|\mathbf{m}, \mathbf{S})$, to the target, $p(\mathbf{w})$, by minimising $\mathrm{KL}(q(\mathbf{w})|p(\mathbf{w}))$ with respect to the moments $\mathbf{m}$ and $\mathbf{S}$. For Gaussian $q(\mathbf{w}) = \mathcal{N}(\mathbf{w}|\mathbf{m}, \mathbf{S})$ and a latent linear model target of the form described in Section 2.3, the G-KL lower-bound on $\log Z$ can be expressed as

$$\mathcal{B}_{G\text{-}KL}(\mathbf{m}, \mathbf{S}) = \underbrace{\frac{1}{2}\log\det(2\pi e\mathbf{S})}_{\text{entropy}} + \underbrace{\sum_{n=1}^{N}\left\langle \log\phi_n(m_n + zs_n)\right\rangle_{\mathcal{N}(z|0,1)}}_{\text{site-projection potentials}} - \underbrace{\frac{1}{2}\left[\log\det(2\pi\boldsymbol{\Sigma}) + (\mathbf{m}-\boldsymbol{\mu})^{\mathsf{T}}\boldsymbol{\Sigma}^{-1}(\mathbf{m}-\boldsymbol{\mu}) + \mathrm{trace}\left(\boldsymbol{\Sigma}^{-1}\mathbf{S}\right)\right]}_{\text{Gaussian potential}}. \quad (4.1.1)$$

Where we show that the site-projection potential expectations $\left\langle \log\phi_n(\mathbf{w}^{\mathsf{T}}\mathbf{h}_n)\right\rangle_{q(\mathbf{w})}$ simplify to the univariate Gaussian expectations $\langle \log\phi_n(m_n + zs_n)\rangle$ in Appendix A.2 following the original presentation made by Barber and Bishop [1998a].



(a) $-|x|$ (b) $-\langle |x| \rangle_{q(x)}$ (c) $-\mathtt{sgn}(x)$ (d) $-\langle \mathtt{sgn}(x) \rangle$

Figure 4.1: Non-differentiable functions and their Gaussian expectations. Figures (a) and (c) plot the non-differentiable function $\psi(x) = -|x|$ and the non-continuous function $\psi(x) = -\mathtt{sgn}(x)$. Figures (b) and (c) plot the expectations of those functions for Gaussian distributed x as a function of the Gaussian mean m *i.e.* $\langle \psi(x) \rangle_{\mathcal{N}(x|m,\sigma^2)}$. The expectations are smooth with respect to the Gaussian mean. As the variance of the Gaussian tends to zero the expectation converges to the underlying function value. Gaussian expectations taken with respect to $\mathcal{N}\left(x|m,\sigma^2\right)$ where $\sigma = 0.0125, 0.5, 1, 2$.

4.2 G-KL bound optimisation

G-KL approximate inference proceeds to obtain the tightest lower-bound to $\log Z$ and the 'closest' Gaussian approximation to $p(\mathbf{w})$ by maximising $\mathcal{B}_{G\text{-}KL}(\mathbf{m}, \mathbf{S})$ with respect to the moments $\mathbf{m}$ and $\mathbf{S}$ of the variational Gaussian density. Therefore, to realise the benefits of G-KL approximate inference we require stable and scalable algorithms to optimise the bound. To this end we now show that for a broad class of models the G-KL objective is both differentiable and concave.

4.2.1 G-KL bound differentiability

Whilst the target density of our model may not be differentiable in $\mathbf{w}$ the G-KL bound with respect to the variational moments $\mathbf{m}, \mathbf{S}$ frequently is. See Figure 4.1 for a depiction of this phenomenon for two, simple, non-differentiable functions. The G-KL bound is in fact smooth for potential functions that are neither differentiable nor continuous (for example they have jump discontinuities). In Appendix B.3 we show that the G-KL bound is smooth for potential functions that are piecewise smooth with a finite number of discontinuities, and where the logarithm of each piecewise segment is a quadratic. This class of functions includes the widely used Laplace density amongst others.

4.2.2 G-KL bound concavity

If each site potential $\{\phi_n\}_{n=1}^N$ is log-concave then the G-KL bound $\mathcal{B}_{G\text{-}KL}(\mathbf{m}, \mathbf{S})$ is jointly concave with respect to the variational Gaussian mean $\mathbf{m}$ and $\mathbf{C}$ the upper-triangular Cholesky decomposition of covariance such that $\mathbf{S} = \mathbf{C}^\mathsf{T}\mathbf{C}$. We say that $f(x)$ is log-concave if $\log f(x)$ is concave in x.

Since the bound depends on the logarithm of $\prod_{n=1}^N \phi_n$ without loss of generality we may take

$N = 1$. Ignoring constants with respect to $\mathbf{m}$ and $\mathbf{C}$, we can write the G-KL bound as

$$\mathcal{B}_{G\text{-}KL}(\mathbf{m}, \mathbf{C}) \overset{\text{c.}}{=} \sum_{d=1}^{D} \log C_{dd} - \frac{1}{2}\mathbf{m}^\mathsf{T}\boldsymbol{\Sigma}^{-1}\mathbf{m} + \boldsymbol{\mu}^\mathsf{T}\boldsymbol{\Sigma}^{-1}\mathbf{m} - \frac{1}{2}\text{trace}\left(\boldsymbol{\Sigma}^{-1}\mathbf{C}\mathbf{C}^\mathsf{T}\right) + \left\langle \log \phi(\mathbf{w}^\mathsf{T}\mathbf{h}) \right\rangle. \quad (4.2.1)$$

Excluding $\left\langle \log \phi(\mathbf{w}^\mathsf{T}\mathbf{h}) \right\rangle$ from the expression above all terms are concave functions exclusively in either $\mathbf{m}$ or $\mathbf{C}$. Since the sum of concave functions on distinct variables is jointly concave the terms in the first line of equation (4.2.1) represent a jointly concave contribution to the bound.

To complete the proof[1] we need to show that $\left\langle \log \phi(\mathbf{w}^\mathsf{T}\mathbf{h}) \right\rangle$ is jointly concave in $\mathbf{m}$ and $\mathbf{C}$. Log-concavity of $\phi(x)$ is equivalent to the statement that for any $x_1, x_2 \in \mathbb{R}$ and any $\theta \in [0, 1]$

$$\log \phi(\theta x_1 + (1-\theta)x_2) \geq \theta \log \phi(x_1) + (1-\theta)\log \phi(x_2). \quad (4.2.2)$$

Therefore, to show that $\mathcal{E}(\mathbf{m}, \mathbf{C}) := \left\langle \log \phi(\mathbf{w}^\mathsf{T}\mathbf{h}) \right\rangle_{\mathcal{N}(\mathbf{w}|\mathbf{m}, \mathbf{C}^\intercal\mathbf{C})}$ is concave it suffices to show for any $\theta \in [0, 1]$ that

$$\mathcal{E}(\theta\mathbf{m}_1 + (1-\theta)\mathbf{m}_2, \theta\mathbf{C}_1 + (1-\theta)\mathbf{C}_2) \geq \theta\mathcal{E}(\mathbf{m}_1, \mathbf{C}_1) + (1-\theta)\mathcal{E}(\mathbf{m}_2, \mathbf{C}_2). \quad (4.2.3)$$

This can be done by making the substitution $\mathbf{w} = \theta\mathbf{m}_1 + (1-\theta)\mathbf{m}_2 + (\theta\mathbf{C}_1 + (1-\theta)\,\mathbf{C}_2)^\mathsf{T}\mathbf{z}$, giving

$$\mathcal{E}\left(\theta\mathbf{m}_1 + (1-\theta)\,\mathbf{m}_2, \theta\mathbf{C}_1 + (1-\theta)\,\mathbf{C}_2\right) = \int \mathcal{N}\left(\mathbf{z}|\mathbf{0}, \mathbf{I}\right) \times \log \phi\left(\theta\mathbf{h}^\mathsf{T}\left(\mathbf{m}_1 + \mathbf{C}_1^\mathsf{T}\mathbf{z}\right) + (1-\theta)\,\mathbf{h}^\mathsf{T}\left(\mathbf{m}_2 + \mathbf{C}_2^\mathsf{T}\mathbf{z}\right)\right) d\mathbf{z}.$$

Using concavity of $\log \phi(x)$ with respect to x and equation (4.2.2) with $\mathbf{w}_1 = \mathbf{m}_1 + \mathbf{C}_1^\mathsf{T}\mathbf{z}$ and $\mathbf{w}_2 = \mathbf{m}_2 + \mathbf{C}_2^\mathsf{T}\mathbf{z}$ we have that

$$\begin{aligned}\mathcal{E}\left(\theta\mathbf{m}_1 + (1-\theta)\,\mathbf{m}_2, \theta\mathbf{C}_1 + (1-\theta)\,\mathbf{C}_2\right) &\geq \theta \int \mathcal{N}\left(\mathbf{z}|\mathbf{0}, \mathbf{I}\right) \log \phi\left(\mathbf{h}^\mathsf{T}\left(\mathbf{m}_1 + \mathbf{C}_1^\mathsf{T}\mathbf{z}\right)\right) d\mathbf{z} \\ &\quad + (1-\theta) \int \mathcal{N}\left(\mathbf{z}|\mathbf{0}, \mathbf{I}\right) \log \phi\left(\mathbf{h}^\mathsf{T}\left(\mathbf{m}_2 + \mathbf{C}_2^\mathsf{T}\mathbf{z}\right)\right) d\mathbf{z} \\ &= \theta\mathcal{E}\left(\mathbf{m}_1, \mathbf{C}_1\right) + (1-\theta)\,\mathcal{E}\left(\mathbf{m}_2, \mathbf{C}_2\right).\end{aligned}$$

Thus the G-KL bound is jointly concave in $\mathbf{m}, \mathbf{C}$ provided all site potentials $\{\phi_n\}_{n=1}^N$ are log-concave.

With consequence to the theoretical convergence rates of gradient based optimisation procedures, the bound is also strongly-concave. A function $f(\mathbf{x})$ is strongly-concave if there exists some $c < 0$ such that for all $\mathbf{x}$, $\nabla^2 f(\mathbf{x}) \preceq c\mathbf{I}$ [Boyd and Vandenberghe, 2004, Section 9.1.2].[2] For the G-KL bound the constant c can be assessed by inspecting the covariance of the Gaussian potential, $\boldsymbol{\Sigma}$. If we arrange the set of all G-KL variational parameters as a vector formed by concatenating $\mathbf{m}$ and the non-zero elements of the column's of $\mathbf{C}$ then the Hessian of $\langle \log \mathcal{N}(\mathbf{w}|\boldsymbol{\mu}, \boldsymbol{\Sigma}) \rangle$ is a block diagonal matrix. Each block of this Hessian is either $-\boldsymbol{\Sigma}^{-1}$ or its submatrix $\left[-\boldsymbol{\Sigma}^{-1}\right]_{i:D, i:D}$, where $i = 2, \ldots, D$. The set of eigenvalues of a block diagonal matrix is the union of the eigenvalues of each of the block matrices' eigenvalues.

[1] This proof was provided by Michalis K. Titsias and simplifies the original presentation made in Challis and Barber [2011], and which is reproduce in Appendix B.7.

[2] We say for square matrices $\mathbf{A}$ and $\mathbf{B}$ that $\mathbf{A} \preceq \mathbf{B}$ iff $\mathbf{B} - \mathbf{A}$ is positive semidefinite.

Furthermore, the eigenvalues of each submatrix are bounded by the upper and lower eigenvalues of $-\boldsymbol{\Sigma}^{-1}$. Therefore $\nabla^2 \mathcal{B}_{G\text{-}KL}(\mathbf{m}, \mathbf{S}) \succeq c\mathbf{I}$ where c is -1 times the smallest eigenvalue of $\boldsymbol{\Sigma}^{-1}$. The sum of a strongly-concave function and a concave function is strongly-concave and thus the G-KL bound as a whole is strongly-concave.

For G-KL bound optimisation using Newton's method to exhibit quadratic convergence rates two additional sufficient conditions, beyond strong concavity and differentiability, need to be shown. The additional requirements being that the G-KL bound has closed sublevel sets and that the G-KL bound's Hessian is Lipschitz continuous on those sublevel sets. For brevity of exposition we present both of these results in Appendix B.3.

4.2.3 Summary

In this section, and in Appendix B.3, we have provided conditions for which the G-KL bound is strongly concave, smooth, has closed sublevel sets and Lipschitz continuous Hessians. Under these conditions optimisation of the G-KL bound will have quadratic convergence rates using Newton's method and super-linear convergence rates using quasi-Newton methods [Nocedal and Wright, 2006, Boyd and Vandenberghe, 2004]. For larger problems, where cubic scaling properties arising from the approximate Hessian calculations required by quasi-Newton methods are infeasible, we will use limited memory quasi-Newton methods, nonlinear conjugate gradients or Hessian free Newton methods to optimise the G-KL bound.

Concavity with respect to the G-KL mean is clear and intuitive – for any fixed G-KL covariance the G-KL bound as a function of the mean can be interpreted as a Gaussian blurring of $\log p(\mathbf{w})$ – see Figure 4.1. As $\mathbf{S} = \nu^2\mathbf{I} \to \mathbf{0}$ then $\mathbf{m}^* \to \mathbf{w}^{MAP}$ where $\mathbf{m}^*$ is the optimal G-KL mean and $\mathbf{w}^{MAP}$ is the maximum a posteriori (MAP) parameter setting.

Another deterministic Gaussian approximate inference procedure applied to the latent linear model class are local variational bounding methods – introduced in Section 3.8. For log-concave potentials local variational bounding methods, which optimise a different criterion with a different parameterisation to the G-KL bound, have also been shown to result in a convex optimisation problem [Seeger and Nickisch, 2011b]. To the best of our knowledge, local variational bounding and G-KL approximate inference methods are the only known concave variational inference procedures for latent linear models as defined in Section 2.3.

Whilst G-KL bound optimisation and MAP estimation share conditions under which they are concave problems, the G-KL objective is often differentiable when the MAP objective is not. Non-differentiable potentials are used throughout machine learning and statistics. Indeed, the practical utility of such non-differentiable potentials in statistical modelling has driven a lot of research into speeding up algorithms to find the mode of these densities – for example see Schmidt et al. [2007]. Despite recent progress these algorithms tend to have slower convergence rates than quasi-Newton methods on smooth, strongly-convex objectives with Lipschitz continuous gradients and Hessians.

One of the significant practical advantages of G-KL approximate inference over MAP estimation and the Laplace approximation is that the target density is not required to be differentiable. With regards

to the complexity of G-KL bound optimisation, whilst an additional cost is incurred over MAP estimation from specifying and optimising the variance of the approximation, a saving is made in the number of times the objective and its gradients need to be computed. Quantifying the net saving (or indeed cost) of G-KL optimisation over MAP estimation is an interesting question reserved for later work.

4.3 Complexity : G-KL bound and gradient computations

In the previous section we provided conditions for which the G-KL bound is strongly concave and differentiable and so provided conditions for which G-KL bound optimisation using quasi-Newton methods will exhibit super-linear convergence rates. Whilst such convergence rates are highly desirable they do not in themselves guarantee that optimisation is scalable. An important practical consideration is the numerical complexity of the bound and gradient computations required by any gradient ascent optimisation procedure.

Discussing the complexity of G-KL bound and gradient evaluations in full generality is complex we therefore restrict ourselves to considering one particularly common case. We consider models where the covariance of the Gaussian potential is spherical, such that $\boldsymbol{\Sigma} = \nu^2\mathbf{I}$. For models that do not satisfy this assumption, in Appendix B.4 we present a full breakdown of the complexity of bound and gradient computations for each G-KL covariance parameterisation presented in Section 4.3.1.3 and a range of parameterisations for the Gaussian potential $\mathcal{N}(\mathbf{w}|\mathbf{m}, \boldsymbol{\Sigma})$.

Note that problems where $\boldsymbol{\Sigma}$ is not a scaling of the identity can be reparameterised to an equivalent problem for which it is. For some problems this reparameterisation can provide significant reductions in complexity. This procedure, the domains for which it is suitable, and the possible computational savings it provides are discussed at further length in Appendix B.5.

For Cholesky factorisations of covariance, $\mathbf{S} = \mathbf{C}^\mathsf{T}\mathbf{C}$, of dimension D the bound and gradient contributions from the $\log\det(\mathbf{S})$ and $\mathrm{trace}(\mathbf{S})$ terms in equation (4.1.1) scale $O(D)$ and $O\left(D^2\right)$ respectively. Terms in equation (4.1.1) that are a function exclusively of the G-KL mean, $\mathbf{m}$, scale at most $O(D)$ and are the cheapest to evaluate. The computational bottleneck arises from the projected variational variances $s_n^2 = \|\mathbf{C}^\mathsf{T}\mathbf{h}_n\|^2$ required to compute each $\left\langle \log\phi_n(\mathbf{w}^\mathsf{T}\mathbf{h}_n)\right\rangle$ term. Computing all such projected variances scales $O\left(ND^2\right)$.[3]

A further computational expense is incurred from computing the N one dimensional integrals required to evaluate $\sum_{n=1}^N \left\langle \log\phi_n(\mathbf{w}^\mathsf{T}\mathbf{h}_n)\right\rangle$. These integrals are computed either numerically or analytically depending on the functional form of ϕ_n. Regardless, this computation scales $O(N)$, possibly though with a significant prefactor. When numerical integration is required, we note that since $\left\langle \log\phi_n(\mathbf{w}^\mathsf{T}\mathbf{h}_n)\right\rangle$ can be expressed as $\left\langle \log\phi_n(m_n + zs_n)\right\rangle_{\mathcal{N}(z|0,1)}$ we can usually assert that the integrand's significant mass lies for $z \in [-5, 5]$ and so that quadrature will yield sufficiently accurate results at modest computational expense. For all the experiments considered here we used fixed width rectangular quadrature and performing these integrals was not the principal bottleneck. For modelling scenarios where this is not the case we note that a two dimensional lookup table can be constructed, at a one off

[3] We note that since a Gaussian potential, $\mathcal{N}(\mathbf{w}|\boldsymbol{\mu}, \boldsymbol{\Sigma})$, can be written as a product over D site-projection potentials computing $\langle \log\mathcal{N}(\mathbf{w}|\boldsymbol{\mu}, \boldsymbol{\Sigma})\rangle$ will in general scale $O\left(D^3\right)$ – see Appendix B.1.3.

cost, to approximate $\langle \log \phi(m + zs) \rangle$ and its derivatives as a function of m and s.

Thus for a broad class of models the G-KL bound and gradient computations scale $O\left(ND^2\right)$ for general parameterisations of the covariance $\mathbf{S} = \mathbf{C}^\mathsf{T}\mathbf{C}$. In many problems of interest the fixed vectors $\mathbf{h}_n$ are sparse. Letting L denote the number of non-zero elements in each vector $\mathbf{h}_n$, computing $\left\{s_n^2\right\}_{n=1}^N$ scales now $O\left(NDL\right)$ where frequently $L \ll D$. Nevertheless, such scaling for the G-KL method can be prohibitive for large problems and so constrained parameterisations are required.

4.3.1 Constrained parameterisations of G-KL covariance

Unconstrained G-KL approximate inference requires storing and optimising $\frac{1}{2}D(D+1)$ parameters to specify the G-KL covariance's Cholesky factor $\mathbf{C}$. In many settings this can be prohibitive. To this end we now consider constrained parameterisations of covariance that reduce both the time and space complexity of G-KL procedures.

Gaussian densities can be parameterised with respect to the covariance or its inverse the precision matrix. A natural question to ask is which of these is best suited for G-KL bound optimisation. Unfortunately, the G-KL bound is neither concave nor convex with respect to the precision matrix. What is more, the complexity of computing the ϕ_n site potential contributions to the bound increases for the precision parameterised G-KL bound. Thus the G-KL bound seems more naturally parameterised in terms of covariance than precision.

4.3.1.1 Optimal G-KL covariance structure

As originally noted by Seeger [1999], the optimal structure for the G-KL covariance can be assessed by calculating the derivative of $\mathcal{B}_{G\text{-}KL}(\mathbf{m}, \mathbf{S})$ with respect to $\mathbf{S}$ and equating it to zero. Doing so, $\mathbf{S}$ is seen to satisfy

$$\mathbf{S}^{-1} = \mathbf{\Sigma}^{-1} + \mathbf{H}\mathbf{\Gamma}\mathbf{H}^\mathsf{T}, \tag{4.3.1}$$

where $\mathbf{H} = [\mathbf{h}_1, \ldots, \mathbf{h}_n]$ and $\mathbf{\Gamma}$ is diagonal such that

$$\Gamma_{nn} = \left\langle \left(z^2 - 1\right) \frac{\log \phi_n(m_n + zs_n)}{2s_n^2} \right\rangle_{\mathcal{N}(z|0,1)}. \tag{4.3.2}$$

$\mathbf{\Gamma}$ depends on $\mathbf{S}$ through the projected variance terms $s_n^2 = \mathbf{h}_n^\mathsf{T}\mathbf{S}\mathbf{h}_n$ and equation (4.3.1) does not provide a closed form expression to solve for $\mathbf{S}$. Furthermore, iterating equation (4.3.1) is not guaranteed to converge to a fixed point or uniformly increase the bound. Indeed this iterative procedure frequently diverges. We are free, however, to directly optimise the bound by treating the diagonal entries of $\mathbf{\Gamma}$ as variational parameters and thus change the number of parameters required to specify $\mathbf{S}$ from $\frac{1}{2}D(D+1)$ to N. This procedure, whilst possibly reducing the number of free parameters, requires us to compute $\log\det(\mathbf{S})$ and $\mathbf{S}$ which in general scales $O\left(ND\min\{D, N\}\right)$ using the matrix inversion lemma – infeasible when $N, D \gg 1$.

A further consequence of using this parameterisation of covariance is that the bound is non-concave. We know from Seeger and Nickisch [2011b] that parameterising $\mathbf{S}$ according to equation (4.3.1) renders $\log\det(\mathbf{S})$ concave with respect to $(\Gamma_{nn})^{-1}$. However the site-projection potentials are not concave with respect to $(\Gamma_{nn})^{-1}$ thus the bound is neither concave nor convex for this parameterisation resulting in

convergence to a possibly local optimum. Non-convexity and $O\left(D^3\right)$ scaling motivates the search for better parameterisations of covariance.

Khan et al. [2012] propose a new technique that uses the decomposition of covariance described in equation (4.3.1) to efficiently optimise the G-KL bound for the special case of Gaussian process regression models. Since for GP regression models $\mathbf{H} = \mathbf{I}_{N\times N}$, the algorithm makes use of the fact that at the optimum of the G-KL bound $\mathbf{S}^{-1}$ differs from $\mathbf{\Sigma}^{-1}$ only at the diagonal elements. The derived fixed point optimisation procedure can potentially speed up G-KL inference in GP models. However, for general latent linear models this approach is not applicable and the need for scalable and general purpose G-KL bound optimisation methods remains.

4.3.1.2 Factor analysis

Parameterisations of the form $\mathbf{S} = \mathbf{\Theta}\mathbf{\Theta}^{\mathsf{T}} + \operatorname{diag}\left(\mathbf{d}^2\right)$ can capture the K leading directions of variance for a $D \times K$ dimensional loading matrix $\mathbf{\Theta}$. Unfortunately this parameterisation renders the G-KL bound non-concave. Non-concavity is due to the entropic contribution $\log\det\left(\mathbf{S}\right)$ which is not even unimodal. All other terms in the bound remain concave under this factorisation. Provided one is happy to accept convergence to possibly local optimum, this is still a useful parameterisation. Computing the projected variances with $\mathbf{S}$ in this form scales $O\left(NDK\right)$ and evaluating $\log\det\left(\mathbf{S}\right)$ and its derivative scales $O\left(K^2(K+D)\right)$.

4.3.1.3 Constrained concave parameterisations

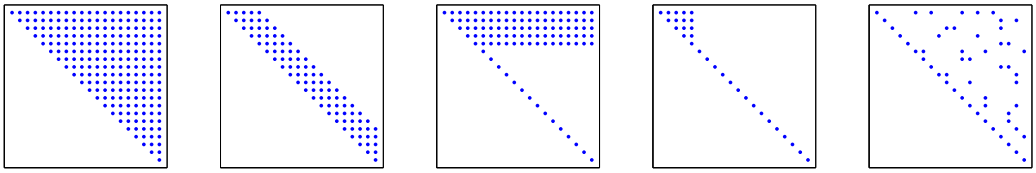
Below we present constrained parameterisations of covariance which reduce both the space and time complexity of G-KL bound optimisation whilst preserving concavity. To reiterate, the computational scaling figures for the bound and gradient computations listed below correspond to evaluating the projected G-KL variance terms, the bottleneck for models with an isotropic Gaussian potential $\mathbf{\Sigma} = \sigma^2\mathbf{I}$. The scaling properties for other models are presented in Appendix B.4. The constrained parameterisations below have different qualities regarding the expressiveness of the variational Gaussian approximation. We note that a zero at the $(i,j)^{th}$ element of covariance specifies a marginal independence relation between parameters w_i and w_j. Conversely, a zero at the $(i,j)^{th}$ element of precision corresponds to a conditional independence relation between parameters w_i and w_j when conditioned on the other remaining parameters.

Banded Cholesky

The simplest option is to constrain the Cholesky matrix to be banded, that is $C_{ij} = 0$ for $j > i+B$ where B is the bandwidth. Doing so reduces the cost of a single bound or gradient computation to $O\left(NDB\right)$. Such a parameterisation describes a sparse covariance matrix and assumes zero covariance between variables that are indexed out of bandwidth. The precision matrix for banded Cholesky factorisations of covariance will not in general be sparse.

Chevron Cholesky

We constrain $\mathbf{C}$ such that $C_{ij} = \Theta_{ij}$ when $j \geq i$ and $i \leq K$, $C_{ii} = d_i$ for $i > K$ and 0 otherwise. We refer to this parameterisation as the chevron Cholesky since the sparsity structure has a broad inverted



(a) Full (b) Banded (c) Chevron (d) Subspace (e) Sparse

Figure 4.2: Sparsity structure for constrained concave Cholesky decompositions of covariance.

'V' shape – see Figure 4.2. Generally, this constrained parameterisation results in a non-sparse covariance but sparse precision. This parameterisation is not invariant to index permutations and so not all covariates have the same representational power. For a Cholesky matrix of this form bound and gradient computations scale $O(NDK)$.

Sparse Cholesky

In general the bound and gradient can be evaluated more efficiently if we impose any fixed sparsity structure on the Cholesky matrix $\mathbf{C}$. In certain modelling scenarios we know a priori which variables are marginally dependent and independent and so may be able construct a sparse Cholesky matrix to reflect that domain knowledge. This is of use in cases where a low band width index ordering cannot be found. For a sparse Cholesky matrix with DK non-zero elements bound and gradient computations scale $O(NDK)$.

Subspace Cholesky.

Another reduced parameterisation of covariance can be obtained by considering arbitrary rotations in parameter space, $\mathbf{S} = \mathbf{E}\mathbf{C}^{\mathsf{T}}\mathbf{C}\mathbf{E}^{\mathsf{T}}$ where $\mathbf{E}$ is a rotation matrix which forms an orthonormal basis over $\mathbb{R}^D$. Substituting this form for the covariance into equation (4.2.1) and for $\mathbf{\Sigma} = \nu^2\mathbf{I}$ we obtain, up to a constant,

$$\mathcal{B}_{G\text{-}KL}(\mathbf{m}, \mathbf{C}) \stackrel{c.}{=} \sum_i \log C_{ii} - \frac{1}{2\nu^2}\left[\|\mathbf{C}\|^2 + \|\mathbf{m}\|^2\right] + \frac{1}{\nu^2}\boldsymbol{\mu}^{\mathsf{T}}\mathbf{m} + \sum_n \left\langle \log \phi(m_n + z s_n)\right\rangle_{\mathcal{N}(z|0,1)},$$

where $s_n = \|\mathbf{C}\mathbf{E}^{\mathsf{T}}\mathbf{h}_n\|$. One may reduce the computational burden by decomposing $\mathbf{E}$ into two submatrices such that $\mathbf{E} = [\mathbf{E}_1, \mathbf{E}_2]$ where $\mathbf{E}_1 \in \mathbb{R}^{D\times K}$ and $\mathbf{E}_w \in \mathbb{R}^{D\times L}$ for $L = (D-K)$. Constraining $\mathbf{C}$ such that $\mathbf{C} = \text{blkdiag}(\mathbf{C}_1, c\mathbf{I}_{L\times L})$, with $\mathbf{C}_1$ a $K \times K$ Cholesky matrix we have that

$$s_n^2 = \|\mathbf{C}_1\mathbf{E}_1^{\mathsf{T}}\mathbf{h}_n\|^2 + c^2(\|\mathbf{h}_n\|^2 - \|\mathbf{E}_1^{\mathsf{T}}\mathbf{h}_n\|^2),$$

meaning that only the K subspace vectors in $\mathbf{E}_1$ are needed to compute $\left\{s_n^2\right\}_{n=1}^N$.

Since $\{\|\mathbf{h}_n\|\}_{n=1}^N$ need to only be computed once the complexity of bound and gradient computations reduces to scaling in K not D. Further savings can be made if we use banded subspace Cholesky matrices: for $\mathbf{C}_1$ having bandwidth B each bound evaluation and associated gradient computation scales $O(NBK)$.

The success of this factorisation depends on how well $\mathbf{E}_1$ captures the leading directions of posterior variance. One simple approach to select $\mathbf{E}_1$ is to use the leading principal components of the 'dataset'

$\mathbf{H}$. Another option is to iterate between optimising the bound with respect to $\{\mathbf{m}, \mathbf{C}_1, c\}$ and $\mathbf{E}_1$. We consider two approaches for optimisation with respect to $\mathbf{E}_1$. The first utilises the form for the optimal G-KL covariance, equation (4.3.2). By substituting in the projected mean and variance terms m_n and s_n^2 into equation (4.3.2) we can set $\mathbf{E}_1$ to be a rank K approximation to this $\mathbf{S}$. The best rank K approximation is given by evaluating the smallest K eigenvectors of $\boldsymbol{\Sigma}^{-1} + \mathbf{H}\boldsymbol{\Gamma}\mathbf{H}^{\mathsf{T}}$. For very large sparse problems we can approximate this using the iterative Lanczos methods described by Seeger and Nickisch [2010]. For smaller non-sparse problems more accurate approximations are available. The second approach is to optimise the G-KL bound directly with respect to $\mathbf{E}_1$ under the constraint that the columns of $\mathbf{E}_1$ are orthonormal. One route to achieving this is to use a projected gradient ascent method.

In Appendix B.1 we provide equations for each term of the G-KL bound and its gradient for each of the covariance parameterisations considered above.

4.4 Comparing Gaussian approximate inference procedures

Due to their favourable computational and analytical properties multivariate Gaussian densities are used by many deterministic approximate inference routines. As discussed in Chapter 3, for latent linear models three popular, deterministic, Gaussian, approximate inference techniques are local variational bounding methods, Laplace approximations and Gaussian expectation propagation. In this section we briefly review and compare the G-KL procedure, as proposed in this chapter, to these other deterministic Gaussian approximate inference methods.

Of the three Gaussian approximate inference methods listed above only one, local variational bounding, provides a lower-bound to the normalisation constant Z. Local variational bounding (LVB) methods were introduced in Section 3.8. In Section 4.4.1 we develop on this presentation and show that the G-KL lower-bound dominates the local lower-bound on $\log Z$.

In Section 4.4.2 we discuss and compare the applicability and computational scaling properties of each deterministic Gaussian approximate inference method presented in Chapter 3 to the G-KL procedure as presented in this chapter.

4.4.1 Gaussian lower-bounds

An attractive property of G-KL approximate inference is that it provides a strict lower-bound on $\log Z$. Lower-bounding procedures are particularly useful for a number of theoretical and practical reasons. The primary theoretical advantage is that it provides concrete exact knowledge about Z and thus also the target density $p(\mathbf{w})$. Thus the tighter the lower-bound on $\log Z$ is the more informative it is. Practically, optimising a lower-bound is often a more numerically stable task than the criteria provided by other deterministic approximate inference methods.

Another well studied route to obtaining a lower-bound for latent linear models are local variational bounding methods. Local variational bounding (LVB) methods were introduced and discussed in Section 3.8. Whilst both G-KL and LVB methods have been discussed in the literature for some time, little work has been done to elucidate the relation between them. Below we prove that G-KL provides a tighter lower-bound on Z than LVB methods.

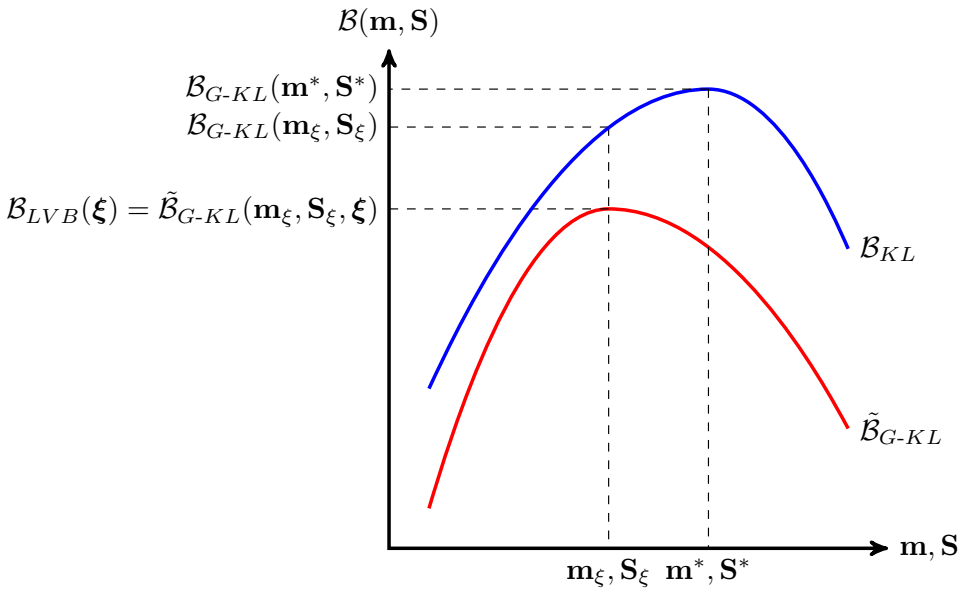


Figure 4.3: Schematic of the relation between the G-KL bound, $\mathcal{B}_{G\text{-}KL}$ (blue), and the weakened KL bound, $\tilde{\mathcal{B}}_{KL}$ (red), plotted as a function of the Gaussian moments $\mathbf{m}$ and $\mathbf{S}$ with $\boldsymbol{\xi}$ fixed. For any setting of the local site bound parameters $\boldsymbol{\xi}$ we have that $\mathcal{B}_{G\text{-}KL}(\mathbf{m}, \mathbf{S}) \geq \tilde{\mathcal{B}}_{KL}(\mathbf{m}, \mathbf{S}, \boldsymbol{\xi})$. We show in the text that the local bound, $\mathcal{B}(\boldsymbol{\xi})$, is the maximum of the weakened KL bound, that is that $\mathcal{B}(\boldsymbol{\xi}) = \max_{\mathbf{m},\mathbf{S}} \tilde{\mathcal{B}}(\mathbf{m}, \mathbf{S}, \boldsymbol{\xi})$ with $\mathbf{m}_\xi, \mathbf{S}_\xi = \operatorname{argmax}_{\mathbf{m},\mathbf{S}} \tilde{\mathcal{B}}(\mathbf{m}, \mathbf{S}, \boldsymbol{\xi})$ in the figure. The G-KL bound can be optimised beyond $\mathcal{B}_{G\text{-}KL}(\mathbf{m}_\xi, \mathbf{S}_\xi)$ to obtain different, optimal G-KL moments $\mathbf{m}^*$ and $\mathbf{S}^*$ that achieve a tighter lower-bound on $\log Z$.

Comparing G-KL and local bounds

An important question is which method, LVB or G-KL, gives a tighter lower-bound on $\log Z$. Each bound derives from a fundamentally different criterion and it is not immediately clear which if either is superior. The G-KL bound has been noted before, empirically in the case of binary classification [Nickisch and Rasmussen, 2008] and analytically for the special case of symmetric potentials [Seeger, 2009], to be tighter than the local bound. It is tempting to conclude that such observed superiority of the G-KL method is to be expected since the G-KL bound has potentially unrestricted covariance $\mathbf{S}$ and so a richer parameterisation. However, many problems have more site potentials ϕ_n than Gaussian moment parameters, that is $N > \frac{1}{2}D(D+3)$, and the local bound in such cases has a richer parameterisation than the G-KL.

We derive a relation between the local and G-KL bounds for $\{\phi_n\}_{n=1}^N$ generic super-Gaussian site potentials. We first substitute the local bound on $\prod_{n=1}^N \phi_n(\mathbf{w}^\mathsf{T}\mathbf{h}_n)$, in equation (3.8.1), into equation (4.1.1) to obtain a new bound

$$\mathcal{B}_{G\text{-}KL}(\mathbf{m}, \mathbf{S}) \geq \tilde{\mathcal{B}}_{G\text{-}KL}(\mathbf{m}, \mathbf{S}, \boldsymbol{\xi}),$$

where

$$2\tilde{\mathcal{B}}_{G\text{-}KL} = -2\left\langle \log q(\mathbf{w}) \right\rangle - \log\det\left(2\pi\boldsymbol{\Sigma}\right) + 2\log c(\boldsymbol{\xi}) - \left\langle (\mathbf{w}-\boldsymbol{\mu})^{\mathsf{T}}\boldsymbol{\Sigma}^{-1}(\mathbf{w}-\boldsymbol{\mu}) \right\rangle - \left\langle \mathbf{w}^{\mathsf{T}}\mathbf{F}(\boldsymbol{\xi})\mathbf{w} \right\rangle + 2\left\langle \mathbf{w}^{\mathsf{T}}\mathbf{f}(\boldsymbol{\xi}) \right\rangle.$$

Using equation (3.8.2) this can be written as

$$\tilde{\mathcal{B}}_{G\text{-}KL} = -\left\langle \log q(\mathbf{w}) \right\rangle - \frac{1}{2}\log\det\left(2\pi\boldsymbol{\Sigma}\right) + \log c(\boldsymbol{\xi}) - \frac{1}{2}\boldsymbol{\mu}^{\mathsf{T}}\boldsymbol{\Sigma}^{-1}\boldsymbol{\mu} - \frac{1}{2}\left\langle \mathbf{w}^{\mathsf{T}}\mathbf{A}\mathbf{w} \right\rangle + \left\langle \mathbf{w}^{\mathsf{T}}\mathbf{b} \right\rangle.$$

By defining $\tilde{q}(\mathbf{w}) = \mathcal{N}\left(\mathbf{w}|\mathbf{A}^{-1}\mathbf{b}, \mathbf{A}^{-1}\right)$ we obtain

$$\tilde{\mathcal{B}}_{G\text{-}KL} = -\mathrm{KL}(q(\mathbf{w})|\tilde{q}(\mathbf{w})) - \frac{1}{2}\log\det\left(2\pi\boldsymbol{\Sigma}\right) + \log c(\boldsymbol{\xi}) - \frac{1}{2}\boldsymbol{\mu}^{\mathsf{T}}\boldsymbol{\Sigma}^{-1}\boldsymbol{\mu} + \frac{1}{2}\mathbf{b}^{\mathsf{T}}\mathbf{A}^{-1}\mathbf{b} - \frac{1}{2}\log\det\left(2\pi\mathbf{A}\right).$$

Since $\mathbf{m}, \mathbf{S}$ only appear via $q(\mathbf{w})$ in the KL term, the tightest bound is given when $\mathbf{m}, \mathbf{S}$ are set such that $q(\mathbf{w}) = \tilde{q}(\mathbf{w})$. At this setting the KL term in $\tilde{\mathcal{B}}_{KL}$ is zero and $\mathbf{m}$ and $\mathbf{S}$ are given by

$$\mathbf{S}_\xi = \left(\boldsymbol{\Sigma}^{-1} + \mathbf{F}(\boldsymbol{\xi})\right)^{-1}, \ \mathbf{m}_\xi = \mathbf{S}_\xi\left(\boldsymbol{\Sigma}^{-1}\boldsymbol{\mu} + \mathbf{f}(\boldsymbol{\xi})\right), \tag{4.4.1}$$

and $\tilde{\mathcal{B}}_{KL}\left(\mathbf{m}_\xi, \mathbf{S}_\xi, \boldsymbol{\xi}\right) = \mathcal{B}\left(\boldsymbol{\xi}\right)$. To reiterate, $\mathbf{m}_\xi$ and $\mathbf{S}_\xi$ maximise $\tilde{\mathcal{B}}_{KL}(\mathbf{m}, \mathbf{S}, \boldsymbol{\xi})$ for any fixed setting of $\boldsymbol{\xi}$. Since $\mathcal{B}_{G\text{-}KL}(\mathbf{m}, \mathbf{S}) \geq \tilde{\mathcal{B}}_{KL}(\mathbf{m}, \mathbf{S}, \boldsymbol{\xi})$ we have that,

$$\mathcal{B}_{G\text{-}KL}(\mathbf{m}_\xi, \mathbf{S}_\xi) \geq \tilde{\mathcal{B}}_{G\text{-}KL}(\mathbf{m}_\xi, \mathbf{S}_\xi, \boldsymbol{\xi}) = \mathcal{B}_{LVB}(\boldsymbol{\xi}).$$

The G-KL bound can be optimised beyond this setting and can achieve an even tighter lower-bound on $\log Z$,

$$\mathcal{B}_{G\text{-}KL}(\mathbf{m}^*, \mathbf{S}^*) = \max_{\mathbf{m},\mathbf{S}} \mathcal{B}_{G\text{-}KL}(\mathbf{m}, \mathbf{S}) \geq \mathcal{B}_{G\text{-}KL}(\mathbf{m}_\xi, \mathbf{S}_\xi).$$

Thus optimal G-KL bounds are provably tighter than both the local variational bound and the G-KL bound calculated using the optimal local bound moments $\mathbf{m}_\xi$ and $\mathbf{S}_\xi$. A graphical depiction of this result is presented in Figure 4.3.

The experimental results presented in Chapter 5 show that the improvement in bound values can be significant. Furthermore, constrained parameterisations of covariance, introduced in Section 4.3, which are required when $D \gg 1$, are also frequently observed to outperform local variational solutions despite the fact that they are not provably guaranteed to do so.

4.4.2 Complexity and model suitability comparison

In Chapter 3 we considered various techniques to perform deterministic approximate inference in the latent linear model class, including the G-KL procedure. Below we reconsider the model applicability, optimisation and scalability properties of the G-KL procedure in light of the contributions made in this chapter.

G-KL approximate inference requires that each site-projection potential has unbounded support on $\mathbb{R}$. Unlike Laplace procedures G-KL is applicable for models with non-differentiable site potentials.

Unlike local variational bounding procedures G-KL does not require the site potentials to be super-Gaussian. In contrast to the Gaussian expectation propagation (G-EP) approximation, which is known to suffer from convergence issues when applied to non log-concave target densities, G-KL procedures optimise a strict lower-bound and convergence is guaranteed when gradient ascent procedures are used.

When $\{\phi_n\}_{n=1}^N$ are log-concave G-KL bound optimisation is a concave problem and we are guaranteed to converge to the global optimum of the G-KL bound. Local variational bounding methods have also been shown to be concave problems in this setting [Nickisch and Seeger, 2009]. However, as we have shown in Section 4.4.1, the optimal G-KL bound to $\log Z$ is provably tighter than the local variational bound.

Exact implementations of G-KL approximate inference require storing and optimising over $\frac{1}{2}D(D+3)$ parameters to specify the Gaussian mean and covariance. The Laplace approximation and mean field bounding methods require storing and optimising over just $O(D)$ parameters. The G-EP approximation and LVB bounding methods require storing and optimising over $O(N)$ variational parameters. Thus the G-KL procedure will often requiring storing and optimising over more variational parameters than these alternative deterministic approximate inference methods. G-KL approximate inference will generally be a more computationally expensive procedure than the MAP and Laplace local approximate methods. However, compared to the G-EP and LVB non-factorising global approximation methods, the computations required to evaluate and optimise the G-KL bound compare favourably. An LVB bound evaluation and parameter update scales $O\left(ND^2\right)$ using the efficient implementation procedures discussed. A full G-EP iteration scales $O\left(ND^2\right)$, where we have assumed for simplicity that $N > D$. Similarly, a singe G-KL bound and gradient evaluation scales $O\left(ND^2\right)$. Thus G-KL procedures whilst defining larger optimisation problems require the evaluation of similarly complex computations. Furthermore, since the G-KL bound is concave for log-concave sites G-KL bound optimisation should be rapid using approximate second order gradient ascent procedures. The results of the next chapter confirm that G-KL procedures are competitive with regards to speed and scalability of approximate inference versus these other, non-factorising global Gaussian approximate inference methods.

Importantly, G-KL procedures can be made scalable by using constrained parameterisations of covariance that do not require making a priori factorisation assumptions on the approximating density $q(\mathbf{w})$. Scalable covariance decompositions for G-KL inference maintain a strict lower-bound on $\log Z$ whereas approximate local bound optimisers do not. G-EP, being a fixed point procedure, has been shown to be unstable when using low-rank covariance approximations and appears constrained to scale $O\left(ND^2\right)$ [Seeger and Nickisch, 2011a].

4.5 Summary

In this chapter we have presented several novel theoretical and practical developments concerning the application of Gaussian Kullback-Leibler (G-KL) approximate inference procedures to the latent linear model class. G-KL approximate inference is seing a resurgence of interest from the research community – see for example: Opper and Archambeau [2009], Ormerod and Wand [2012], Honkela et al. [2010], Graves [2011]. The work presented in this chapter provides further justification for its use.

G-KL approximate inference's primary strength over other deterministic Gaussian approximate inference methods is the ease with which it can be applied to new models. All that is required to apply the G-KL method to a target density in the form of equation (2.3.1) is that each potential has unbounded support and that univariate Gaussian expectations of the log of the potential, $\langle \log \phi(z) \rangle_{\mathcal{N}(z|m,s)}$, can be efficiently computed. For most potentials of interest this is equivalent to requiring that the pointwise evaluation of the univariate functions $\{\log \phi_n(z)\}$ can be efficiently computed. Notably, implementing the G-KL procedure for a new potential function ϕ does not require us to derive its derivatives, lower-bounds on it, or complicated update equations. Neither does the procedure place restrictive conditions on the form of the potential functions, for example that it is log-concave, super-Gaussian or differentiable. Furthermore, since the G-KL method optimises a strict lower-bound G-KL approximate inference is found to be numerically stable.

A long perceived disadvantage of G-KL approximate inference is the difficulty of optimising the bound with respect to the Gaussian covariance. Previous authors have advocated optimising the bound with respect to either the full covariance matrix $\mathbf{S}$ or with respect to a particular structured form of covariance that is defined in Section 4.3.1.1. However, using either of these parameterisations renders the bound non-concave and requires multiple cubic matrix operations to evaluate the bound and its derivatives. In this chapter we have shown that using the Cholesky parameterisation of G-KL covariance both reduces the complexity of single bound/derivative evaluations and results in a concave optimisation problem for log-concave sites $\{\phi_n\}_{n=1}^N$. Furthermore, for larger problems we have provided concave constrained parameterisations of covariance that make G-KL methods fast and scalable without resorting to making fully factorised approximations of the target density.

Limited empirical studies have reported that G-KL approximate inference can be one of the most accurate deterministic, global, Gaussian, approximate inference methods considered here. The most closely related global Gaussian deterministic approximate inference method is the local variational bounding procedure since both methods provide a principled lower-bound to the target densities normalisation constant. However, as we showed in Section 4.4.1, G-KL procedures are guaranteed to provide a lower-bound on $\log Z$ that is tighter than LVB methods. Furthermore, in log-concave models, since the G-KL bound is concave, we are guaranteed to find the global optimum of the G-KL bound.

4.6 Future work

As detailed in Section 2.3 we want any deterministic approximate inference routine to be widely applicable, fast and accurate. The work presented in this chapter provided techniques and results that show that G-KL approximate inference (relative to other Gaussian approximate inference methods when applied to latent linear models) can be made accurate and fast. In this section, we consider directions of research to develop the G-KL procedure in terms of its generality, its accuracy or its speed. The generality of G-KL inference can be improved by developing methods to apply the technique to inference problems beyond the latent linear model class. The accuracy of G-KL inference can be improved by expanding the class of variational approximating densities beyond the multivariate Gaussian. The speed and scalability of G-KL inference can by improved by developing new numerical techniques to optimise the G-KL bound.

4.6.1 Increasing the generality

Increasing the generality of the G-KL approximate inference procedure refers to increasing the class of inference problems to which G-KL methods can be successfully and efficiently applied. Below we consider the problem of extending the G-KL approximation method to perform inference in the bilinear model class.

Bilinear models

The latent linear model class describes a conditional relation between the variables we wish to predict/model y, some fixed vector $\mathbf{x}$, and the latent variables $\mathbf{w}$ in the form $y = f(\mathbf{w}^\mathsf{T}\mathbf{x}) + \epsilon$, where f is some non-linear function and ϵ some additive noise term. One extension to this model is to consider bilinear models such that

$$y = f(\mathbf{u}^\mathsf{T}\mathbf{X}\mathbf{v}) + \epsilon, \tag{4.6.1}$$

where we now have two sets of parameters/latent variables $\mathbf{u} \in \mathbb{R}^{D_u}$ and $\mathbf{v} \in \mathbb{R}^{D_v}$, where the matrix $\mathbf{X} \in \mathbb{R}^{D_u \times D_v}$ is fixed. Examples of this model class include popular matrix factorisation models [Seeger and Bouchard, 2012, Salakhutdinov and Mnih, 2008], models to disambiguate style and content [Tenenbaum and Freeman, 2000] and Bayesian factor analysis models where we want to approximate the full posterior on both the factor loading vectors and the latent variables [Tipping and Bishop, 1999]. Often, the MAP approximation is used in this model class since the problem is analytically intractable and the datasets tend to be large. Since the MAP approximation can be quite inaccurate, see the discussion presented in Section 3.3, it is an important avenue of research to develop more accurate yet scalable inference procedures in this model class.

To perform G-KL approximate inference in this model class we would need to optimise the KL divergence $\mathrm{KL}(q(\mathbf{u},\mathbf{v})|p(\mathbf{u},\mathbf{v}|y))$ with respect to $q(\mathbf{u},\mathbf{v})$ a multivariate Gaussian. Re-arranging the KL we can obtain the familiar lower-bound on the normalisation constant of $p(\mathbf{u},\mathbf{v}|y)$ such that

$$\log p(y) \geq H[q(\mathbf{u},\mathbf{v})] + \langle \log p(\mathbf{u}) \rangle_{q(\mathbf{u})} + \langle \log p(\mathbf{v}) \rangle_{q(\mathbf{v})} + \left\langle \log \phi(\mathbf{u}^\mathsf{T}\mathbf{X}\mathbf{v}) \right\rangle_{q(\mathbf{u},\mathbf{v})}, \tag{4.6.2}$$

where we have assumed that the prior/latent densities on $\mathbf{u},\mathbf{v}$ are independent. For Gaussian $q(\mathbf{u},\mathbf{v})$, the difficulty in evaluating and optimising equation (4.6.2) with respect to $q(\mathbf{u},\mathbf{v})$ is due to the energy term $\left\langle \log \phi(\mathbf{u}^\mathsf{T}\mathbf{X}\mathbf{v}) \right\rangle$. Constraining the Gaussian approximation to factorise so that $q(\mathbf{u},\mathbf{v}) = q(\mathbf{u})q(\mathbf{v})$, we see that the energy will not simplify to a univariate Gaussian expectation since $z := \mathbf{u}^\mathsf{T}\mathbf{X}\mathbf{v}$ is not Gaussian distributed. However, we note that if $\phi(\cdot)$ is a exponentiated quadratic function its expectation will admit a simply analytic form [Lim and Teh, 2007].

Therefore, one direction for future work would be to try to construct methods that provide efficient, possibly approximate, evaluation of the energy term in equation (4.6.2). Possible routes to achieve this include: approximately computing the expectation and its derivatives using sampling methods improving on the techniques described in Graves [2011], Blei et al. [2012], bounding the non-Gaussian potential ϕ by a function whose expectation can be computed making a more accurate approximation than is proposed by Seeger and Bouchard [2012], Khan et al. [2010], or by developing numerical techniques

to compute the density of $z := \mathbf{u}^{\mathsf{T}}\mathbf{X}\mathbf{v}$ exactly – for example by adapting the methods considered in Chapter 6.

4.6.2 Increasing the speed

In this section we consider two possible methods that could increase the speed of convergence for G-KL bound optimisation. First, we consider a method that could possibly increase the speed of convergence in moderately sized models. Second, we consider a method to possibly obtain distributed or parallel optimisation of the G-KL objective suitable for much larger problems than previously considered.

Convergent fixed points for $\mathbf{S}$

Honkela et al. [2010] proposed a method to use the local curvature of the KL divergence as a natural gradient pre-conditioner for non-linear conjugate gradient optimisation of the G-KL objective with respect to the Gaussian mean $\mathbf{m}$. The authors reported that this procedure provided faster convergence in a Bayesian Gaussian mixture model and a Bayesian non-linear state space model compared with G-KL bound optimisation using non-linear conjugate gradients. Our own experiments suggest that these experiments do not offer considerable improvements over standard conjugate gradients methods, LBFGs or Hessian free Newton methods for log-concave latent linear models. Presumably this is because the natural gradient preconditioner does not provide significant additional information about the KL objective surface for the simpler, strongly-concave lower-bound surfaces we consider in the latent linear model class. Honkela et al. [2010] optimise $\mathbf{S}$ using the recursion defined in equation (4.3.1) which we have observed to occasionally result in oscillatory, non-convergent updates.

One direction for future work is to try to develop a fixed point iterative procedure for $\mathbf{S}$ by augmenting the recursion in equation (4.3.1). Possibly, convergence could be imposed by damping the update. One possible damping procedure could be to use $\mathbf{\Gamma}_{new} := \theta\mathbf{\Gamma}_{old} + (1-\theta)\mathbf{\Gamma}$ with $\theta \in (0,1)$ and $\mathbf{\Gamma}$ defined as in equation (4.3.2). Another avenue of research would be to derive conditions under which the fixed point is guaranteed to increase the bound. Using these conditions one could possibly construct an optimisation procedure that switches between gradient ascent updates and the fixed point updates. Such a procedure is limited to problems of moderate dimensionality since the fixed point update requires a matrix inversion.

Dual decomposition for distributed optimisation

Modern applications of machine learning and statistics are posing ever larger inference problems. For example, Graepel et al. [2010] develop a Bayesian logistic regression model to drive advertisement click prediction on the web. In this problem the feature set size D and the number of training instances N can be of the order of 10^9. Posterior inference has benefits over point estimation techniques such as the MAP approximations in this problem domain since the posterior can be used to drive on-line exploration and active learning approaches, using for example Thompson sampling methods [Chapelle and Li, 2011]. Typically, inference in problems of this dimensionality is facilitated by placing strong factorisation constraints on the approximating density. However, it may be beneficial to approximate posterior covariance in such problems since this would allow us to derive more accurate (and hence less

costly) exploration strategies. One approach to scaling the G-KL procedure to problems of this size could be to develop distributed optimisation methods.

Following the notation set out earlier in this chapter, the G-KL lower-bound for a model with a spherical zero mean Gaussian potential, $\mathcal{N}\left(\mathbf{w}|\mathbf{0}, \sigma^2\mathbf{I}_D\right)$, and N non-Gaussian site potentials can be expressed as

$$\mathcal{B}_{G\text{-}KL}(\mathbf{m}, \mathbf{C}) \overset{\text{c.}}{=} \sum_{i=1}^{D} \log C_{ii} - \frac{1}{2\sigma^2}\sum_{i=1}^{D} m_i^2 - \frac{1}{2\sigma^2}\sum_{i,j\geq i}^{D} C_{ij}^2 + \sum_{n=1}^{N}\left\langle \log \phi_n(\mathbf{w}^\mathsf{T}\mathbf{h}_n)\right\rangle, \qquad (4.6.3)$$

where we have omitted constants with respect to the variational parameters $\mathbf{m}, \mathbf{C}$. As we can see in equation (4.6.3), excluding the site potential's contribution, the G-KL bound is separable with respect to the variational parameters $\mathbf{m} = \{m_i\}_{i=1}^D$ and $\mathbf{C} = \{C_{ij}\}_{i,j\geq i}$. The complication in developing a parallel optimisation technique for the objective described in equation (4.6.3) is due to the site potential energy terms $\left\langle \log \phi_n(\mathbf{w}^\mathsf{T}\mathbf{h}_n)\right\rangle$. However, as we have shown previously, these terms, alongside the separable entropy and Gaussian potential contribution's to the bound, are concave. Efficient distributed algorithms, for example dual decomposition techniques and the alternating direction method of multipliers (ADMM), have been developed for optimising objectives of this form – see Boyd et al. [2011] for a comprehensive review of such techniques. Thus one possibly fruitful direction for future work would be to adapt methods such as ADMM, which are typically used for MAP estimation problems, to drive distributed optimisation of the G-KL bound. Indeed, recently Khan et al. [2013] have proposed a dual formulation of the G-KL objective that affords a more scalable parallel optimisation procedure.

4.6.3 Increasing the accuracy

G-KL approximate inference is feasible since for Gaussian $q(\mathbf{w})$ both the entropy and the energy terms of the KL bound can be efficiently computed. For the latent linear model class, the energy terms can be efficiently computed for Gaussian $q(\mathbf{w})$ since the D-dimensional expectation $\left\langle \log \phi(\mathbf{w}^\mathsf{T}\mathbf{h})\right\rangle_{q(\mathbf{w})}$ can be simplified to the univariate expectation $\left\langle \log \phi(y)\right\rangle_{q(y)}$ where $q(y)$ is a known, and cheap to evaluate, density – specifically a univariate Gaussian. A natural question to ask then, is for what other density classes $q(\mathbf{w})$ can we express $\left\langle \log \phi(\mathbf{w}^\mathsf{T}\mathbf{h})\right\rangle_{q(\mathbf{w})} = \left\langle \log \phi(y)\right\rangle_{q(y)}$ where $q(y)$ can be efficiently computed? In the next chapter we address this question quite generally by performing marginal inferences in the Fourier domain. Here we consider another density class for which this property might also hold.

Elliptically contoured variational densities

One possible route to generalising the class of approximating densities $q(\mathbf{w})$ is to consider elliptically contoured multivariate densities constructed as a univariate scale mixture of a multivariate Gaussian. Following Eltoft et al. [2006a,b], we define a univariate Gaussian scale mixture as

$$q(\mathbf{w}|\mathbf{m}, \mathbf{S}, \boldsymbol{\rho}) = \int \mathcal{N}\left(\mathbf{w}|\mathbf{m}, \alpha\mathbf{S}\right) p(\alpha|\boldsymbol{\rho}) d\alpha, \qquad (4.6.4)$$

where α is a positive, real-valued random variable with density function $p(\alpha|\boldsymbol{\rho})$. One candidate for $p(\alpha|\boldsymbol{\rho})$ is the Gamma density, in which case equation (4.6.4) is known as the multivariate K distribution. Since the variance scale weighting is univariate, equation (4.6.4) describes a family of densities with elliptic contours.

KL approximate inference could then be generalised beyond simple Gaussian approximations provided the KL divergence $\mathrm{KL}(q(\mathbf{w}|\mathbf{m},\mathbf{S},\boldsymbol{\rho})|p(\mathbf{w}))$ can be evaluated and optimised with respect to the variational parameters $\{\mathbf{m},\mathbf{S},\boldsymbol{\rho}\}$. This would require that we can develop simple efficient routines to compute the energy, the entropy and both of their derivatives for elliptically contoured $q(\mathbf{w}|\mathbf{m},\mathbf{S},\boldsymbol{\rho})$ as defined in equation (4.6.4).

A single energy term, for $q(\mathbf{w})$ as defined in equation (4.6.4), can be expressed as

$$\begin{aligned}\left\langle \log \phi(\mathbf{w}^{\mathsf{T}}\mathbf{h}) \right\rangle_{q(\mathbf{w})} &= \int\int \mathcal{N}\left(\mathbf{w}|\mathbf{m},\alpha\mathbf{S}\right) p(\alpha|\boldsymbol{\rho})\psi(\mathbf{w}^{\mathsf{T}}\mathbf{h})d\mathbf{w}d\alpha \\ &= \int\int \mathcal{N}\left(z|0,\alpha\right) p(\alpha|\boldsymbol{\rho})d\alpha\psi(m+zs)dz \\ &= \int p(z|\boldsymbol{\rho})\psi(m+zs)dz,\end{aligned}$$

where $m := \mathbf{m}^{\mathsf{T}}\mathbf{h}$, $s^2 := \mathbf{h}^{\mathsf{T}}\mathbf{S}\mathbf{h}$ and $\psi := \log\phi$. Thus the multivariate expectation $\left\langle \psi(\mathbf{w}^{\mathsf{T}}\mathbf{h}) \right\rangle_{q(\mathbf{w})}$ can be expressed as a univariate expectation with respect to the marginal $p(z) := \int \mathcal{N}\left(z|0,\alpha\right) p(\alpha|\boldsymbol{\rho})d\alpha$. For these energy terms to be efficiently computable we need to construct a representation of $p(\alpha|\boldsymbol{\rho})$ such that the density $p(z|\boldsymbol{\rho})$ can also be efficiently computed.

The entropy for the Gaussian scale mixture can be decomposed as $H[q(\mathbf{w}|\mathbf{m},\mathbf{S},\boldsymbol{\rho})] = \log\det(\mathbf{S}) + H[q(\mathbf{v}|\boldsymbol{\rho})]$, where $H[q(\mathbf{v}|\boldsymbol{\rho})]$ is the entropy of the 'standard normal' scale mixture $q(\mathbf{v}|\boldsymbol{\rho}) := \int \mathcal{N}\left(\mathbf{v}|\mathbf{0},\alpha\mathbf{I}\right) p(\alpha|\boldsymbol{\rho})d\alpha$. Therefore, we additionally require a method to efficiently compute, or bound, $H[q(\mathbf{v}|\boldsymbol{\rho})]$ to make this procedure practical.

Chapter 5

Gaussian KL approximate inference : experiments

In this chapter we seek to validate the analytical results presented previously by measuring and comparing the numerical performance of the Gaussian KL approximate inference method to other deterministic Gaussian approximate inference routines. Results are presented for three popular machine learning models. In Section 5.1 we compare deterministic Gaussian approximate inference methods in robust Gaussian process regression models. In Section 5.2 we asses the performance of the constrained parameterisations of G-KL covariance that were presented in Section 4.3.1 to perform inference in large scale Bayesian logistic regression models. In light of this, in Section 5.3 we compare the performance of constrained covariance G-KL methods and fast approximate local variational bounding methods in three, large-scale, real world, Bayesian logistic regression models. Finally, in Section 5.4 we compare Gaussian approximate inference methods to drive sequential experimental design procedures in Bayesian sparse linear models.

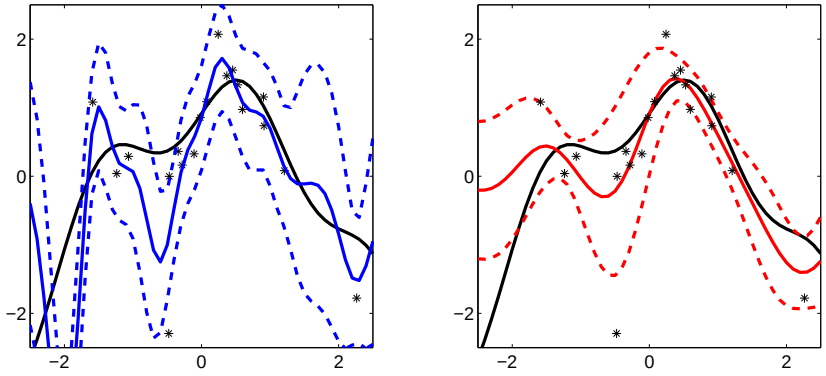
5.1 Robust Gaussian process regression

Gaussian Processes (GP) are a popular non-parametric approach to supervised learning problems, see Rasmussen and Williams [2006] for a thorough introduction, for which inference falls into the general latent linear model form described in Section 2.3. Excluding limited special cases, computing Z and evaluating the posterior density, necessary to make predictions and set hyperparameters, is analytically intractable.

The supervised learning model for fully observed covariates $\mathbf{X} \in \mathbb{R}^{N \times D}$ and corresponding dependent variables $\mathbf{y} \in \mathbb{R}^N$ is specified by the GP prior on the latent function values $\mathbf{w} \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ and the likelihood $p(\mathbf{y}|\mathbf{w})$. The GP prior moments are constructed by the GP covariance and mean functions which take the covariates $\mathbf{X}$ and a vector of hyperparameters $\boldsymbol{\theta}$ as arguments. The posterior on the latent function values, $\mathbf{w}$, is given by

$$p(\mathbf{w}|\mathbf{y}, \mathbf{X}, \boldsymbol{\theta}) = \frac{1}{Z} p(\mathbf{y}|\mathbf{w}) \mathcal{N}(\mathbf{w}|\boldsymbol{\mu}, \boldsymbol{\Sigma}) . \tag{5.1.1}$$

The likelihood factorises over data instances, $p(\mathbf{y}|\mathbf{w}) = \prod_{n=1}^{N} \phi(w_n)$, thus the GP posterior is of the form of equation (2.3.1).



(a) Gaussian likelihood (b) Student's t likelihood

Figure 5.1: Gaussian process regression with a squared exponential covariance function and (a) a Gaussian or (b) a Student's t likelihood. Covariance hyperparameters are optimised for a training dataset with outliers. Latent function posterior mean (solid) and ± 1 standard deviation (dashed) values are plotted in blue (a) and red (b). The data generating function is plotted in black. The Student's t model makes more conservative interpolated predictions whilst the Gaussian model appears to over-fit the data.

GP regression

For GP regression models the likelihood is most commonly Gaussian distributed, equivalent to assuming zero mean additive Gaussian noise. This assumption leads to analytically tractable, indeed Gaussian, forms for the posterior. However, Gaussian additive noise is a strong assumption to make, and is often not corroborated by real world data. Gaussian distributions have thin tales – the density function rapidly tends to zero for values far from the mean – see Figure 2.6. Outliers in the training set then do not have to be too extreme to negatively affect test set predictive accuracy. This effect can be especially severe for GP models that have the flexibility to incorporate training set outliers to areas of high likelihood – essentially over-fitting the data.

An example of GP regression applied to a dataset with outliers is presented in figure 5.1(a). In this figure a GP prior with squared exponential covariance function coupled with a Gaussian likelihood over-fits the training data and the resulting predicted values differ significantly from the underlying data generating function.

One approach to prevent over-fitting is to use a likelihood that is robust to outliers. Heavy tailed likelihood densities are robust to outliers in that they do not penalise too heavily observations far from the latent function mean. Two distributions are often used in this context: the Laplace otherwise termed the double exponential, and the Student's t. The Laplace probability density function can be expressed as

$$p(y|\mu,\tau) = \frac{1}{2\tau} e^{-|y-\mu|/\tau},$$

where τ controls the variance of the random variable x with $\mathrm{var}(y) = 2\tau^2$. The Student's t probability

density function can be written as

$$p(y|\mu,\nu,\sigma^2) = \frac{\Gamma\left(\frac{1}{2}(\nu+1)\right)}{\Gamma\left(\frac{1}{2}\nu\right)\sqrt{\pi\nu\sigma^2}}\left(1+\frac{(y-\mu)^2}{\nu\sigma^2}\right)^{-\frac{\nu+1}{2}}$$

where $\nu \in \mathbb{R}^+$ is the degrees of freedom parameter, $\sigma \in \mathbb{R}^+$ the scale parameter, and $\mathrm{var}(y) = \sigma^2\nu/(\nu-2)$ for $\nu > 2$. As the degrees of freedom parameter becomes increasingly large the Student's t distribution converges to the Gaussian distribution. See Figure 2.6 for a comparison of the Student's t, Laplace and Gaussian density functions.

GP models with outlier robust likelihoods such as the Laplace or the Student's t can yield significant improvements in test set accuracy versus Gaussian likelihood models [Vanhatalo et al., 2009, Jylanki et al., 2011, Opper and Archambeau, 2009]. In figure 5.1(b) we model the same training data as in figure 5.1(a) but with a heavy tailed Student's t likelihood, the resulting predictive values are more conservative and lie closer to the true data generating function than for the Gaussian likelihood model.

Approximate inference

Whilst Laplace and Student's t likelihoods can successfully 'robustify' GP regression models to outliers they also render inference analytically intractable and approximate methods are required. In this section we compare G-KL approximate inference to other deterministic Gaussian approximate inference methods, namely: the Laplace approximation (Lap), local variational bounding (LVB) and Gaussian expectation propagation (G-EP).

Each approximate inference method cannot be applied to each likelihood model. Since the Laplace likelihood is not differentiable everywhere Laplace approximate inference is not applicable. Since the Student's t likelihood is not log-concave, indeed the posterior can be multi-modal, vanilla G-EP implementations are numerically unstable [Seeger et al., 2007]. Recent work [Jylanki et al., 2011] has alleviated some of G-EP's convergence issues for Student's t GP regression, however, these extensions are beyond the scope of this work.

Local variational bounding and G-KL procedures are applied to both likelihood models. For local variational bounding, both the Laplace and Student's t densities are super-Gaussian and thus tight exponentiated quadratic lower-bounds exist – see Seeger and Nickisch [2010] for the precise forms that are employed in these experiments. Laplace, local variational bounding and G-EP results are obtained using the `GPML` toolbox [Rasmussen and Nickisch, 2010].[1] G-KL approximate inference is straightforward, for the G-KL approximate posterior $q(\mathbf{w}) = \mathcal{N}(\mathbf{w}|\mathbf{m},\mathbf{S})$ the likelihood's contribution to the bound is

$$\langle \log p(\mathbf{y}|\mathbf{w})\rangle_{q(\mathbf{w})} = \sum_n \left\langle \log \phi_n(m_n + z\sqrt{S_{nn}})\right\rangle_{\mathcal{N}(z|0,1)}. \tag{5.1.2}$$

Thus the equation above is of the standard site-projection potential form but with $\mathbf{h}_n = \mathbf{e}_n$ the unit norm basis vector and ϕ_n the likelihood of the n^{th} data point. The expectations for the Laplace likelihood site potentials have simple analytic forms – see Appendix A.5.2. The expectations for the Student's t site potentials are evaluated numerically. All other terms in the G-KL bound have simple analytic forms and

[1] The `GPML` toolbox can be downloaded from `www.gaussianprocess.org`.

		Gauss	Student's t			Laplace		
		Exact	G-KL	LVB	Lap	G-KL	LVB	G-EP
C. ST	LML	-15 ± 2	-75 ± 2	-240 ± 21	-7 ± 1	8 ± 5	2 ± 2	$--\pm--$
	MSE	1.15 ± 0.2	1.6 ± 0.2	23.8 ± 4	2.2 ± 0.4	1.3 ± 1.1	1.2 ± 1.0	$--\pm--$
	TLP	0.79 ± 0.10	0.73 ± 0.05	-0.65 ± 0.06	0.41 ± 0.03	0.97 ± 0.06	0.91 ± 0.05	$--\pm--$
Friedman	LML	70 ± 6	-159 ± 7	-578 ± 34	-97 ± 4	-69 ± 6	-73 ± 8	$--\pm--$
	MSE	10 ± 3	5 ± 1	17 ± 2	13 ± 1	5 ± 1	3 ± 1	$--\pm--$
	TLP	-0.26 ± 0.09	0.12 ± 0.09	-0.54 ± 0.06	-0.65 ± 0.06	0.07 ± 0.09	0.25 ± 0.11	$--\pm--$
Neal	LML	39 ± 10	-171 ± 14	-962 ± 1	-21 ± 15	-26 ± 9	-27 ± 8	-14 ± 7
	MSE	1.7 ± 0.6	2.9 ± 1.1	4.4 ± 1.3	0.9 ± 0.5	0.9 ± 0.4	0.9 ± 0.4	0.9 ± 0.5
	TLP	0.22 ± 0.12	0.88 ± 0.03	0.36 ± 0.02	0.67 ± 0.08	0.86 ± 0.04	1.13 ± 0.02	0.91 ± 0.04
Boston	LML	51 ± 3	-133 ± 13	-551 ± 37	-53 ± 3	-60 ± 3	-61 ± 3	-53 ± 4
	MSE	26 ± 1	25 ± 2	26 ± 1	23 ± 2	25 ± 2	26 ± 1	22 ± 1
	TLP	-0.74 ± 0.07	-0.44 ± 0.03	-0.58 ± 0.03	-0.44 ± 0.03	-0.52 ± 0.06	-0.51 ± 0.02	-0.46 ± 0.03

Table 5.1: Gaussian process regression results for different (approximate) inference procedures, likelihood models and datasets. First column section: Gaussian likelihood results with exact inference. Second column section: Student's t likelihood results with G-KL, local variational bounding (LVB) and Laplace (Lap.) approximate inference. Third column section: Laplace likelihood results with G-KL, LVB and Gaussian expectation propagation (G-EP) approximate inference. Each row presents the (approximate or lower-bound) log marginal likelihood (LML), test set mean squared error (MSE), or approximate test set log probability (TLP) values obtained by dataset. Table values are the mean and standard error of the values obtained over the 10 random partitions of the data.

computations that scale $\leq O\left(N^3\right)$. G-KL results are obtained, as for all other results in this paper, using the `vgai` Matlab package – see Appendix B.8. For the Laplace likelihood model, which is log-concave, Hessian free Newton methods were used to optimise the G-KL bound. For the Student's t likelihood, which is not log-concave, LBFGS was used to optimise the G-KL bound.

Experimental setup

We consider GP regression with training data $\mathcal{D} = \{(y_n, \mathbf{x}_n)\}_{n=1}^N$ for covariates $\mathbf{x}_n \in \mathbb{R}^D$ and dependent variables $y_n \in \mathbb{R}$. We assume a zero mean Gaussian process prior on the latent function values, $\mathbf{w} = [w_1, ..., w_N]^\mathsf{T} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\Sigma})$. The covariance, $\boldsymbol{\Sigma}$, is constructed as the sum of the squared exponential kernel and the independent white noise kernel,

$$\Sigma_{mn} = k(\mathbf{x}_m, \mathbf{x}_n, \boldsymbol{\theta}) = \sigma_{se}^2 e^{-\sum_d (x_{nd} - x_{md})^2 / l_d^2} + \gamma^2 \delta(n, m), \tag{5.1.3}$$

where x_{nd} refers to the d^{th} element of the n^{th} covariate, σ_{se}^2 is the 'signal variance' hyperparameter, l_d the squared exponential 'length scale' hyperparameter, and γ the independent white noise hyperparameter (above $\delta(x, y)$ is the Kronecker delta such that $\delta(n, m) = 1$ if $n = m$ and 0 otherwise). Covariance hyperparameters are collected in the vector $\boldsymbol{\theta}$.

We follow the evidence maximisation or maximum likelihood two (ML-II) procedure to estimate the covariance hyperparameters, that is we set covariance hyperparameters to maximise $p(\mathbf{y}|\mathbf{X}, \boldsymbol{\theta})$. Since $p(\mathbf{y}|\mathbf{X}, \boldsymbol{\theta})$ cannot be evaluated exactly we use the approximated values offered by each of the approximate inference methods. Covariance hyperparameters are optimised numerically using nonlinear conju-

gate gradients. The marginal likelihood, $p(\mathbf{y}|\mathbf{X}, \boldsymbol{\theta})$, is not unimodal and we are liable to converge to a local optima regardless of which inference method is used. All methods were initialised with the same hyperparameter setting. Hyperparameter derivatives for the G-KL bound are presented in Appendix B.6.2.

Likelihood hyperparameters were selected to maximise test set log probability scores on a held out validation dataset. Simultaneous likelihood and covariance ML-II hyperparameter optimisation for the Student's t and Laplace likelihoods yielded poor test set performance regardless of the approximate inference method used (as has been previously reported for Student's t likelihoods in other experiments [Vanhatalo et al., 2009, Jylanki et al., 2011]). For the Student's t likelihood model the d.o.f. parameter was fixed with $\nu = 3$.

Results were obtained for the four approximate inference procedures on the four datasets using both the Laplace and the Student's t likelihoods. Two UCI datasets were used:[2] Boston housing and Concrete Slump Test. And two synthetic datasets: Friedman[3] and Neal.[4] Each experiment was repeated over 10 randomly assigned training, validation and test set partitions. The size of each dataset is as follows: Concrete Slump Test $D = 9$, $N_{trn} = 50$, $N_{val} = 25$, $N_{tst} = 28$; Boston $D = 13$, $N_{trn} = 100$, $N_{val} = 100$, $N_{tst} = 306$; Friedman $D = 10$, $N_{trn} = 100$, $N_{val} = 100$, $N_{tst} = 100$; Neal $D = 1$, $N_{trn} = 100$, $N_{val} = 100$, $N_{tst} = 100$. Each partition of the data was normalised using the mean and standard deviation statistics of the training data.

To asses the validity of the Student's t and Laplace likelihoods we also implemented GP regression with a Gaussian likelihood and exact inference.

Results

Results are presented in Table 5.1. Approximate Log Marginal Likelihood (LML), test set Mean Squared Error (MSE) and approximate Test set Log Probability (TLP) mean and standard error values obtained over the 10 partitions of the data are provided. It is important to stress that the TLP values are approximate values for all methods, obtained by summing the approximate log probability of each test point using the surrogate score presented in Appendix B.6.1. For G-KL and LVB procedures the TLP values are not lower-bounds.

The results confirm the utility of heavy tailed likelihoods for GP regression models. Test set predictive accuracy scores are higher with robust likelihoods and approximate inference methods than with a Gaussian likelihood and exact inference. This is displayed in the lower MSE error and higher TLP scores of the best performing robust likelihood results than for the Gaussian likelihood. Exact inference for the Gaussian likelihood model achieves the greatest LML in all problems except the Concrete Slump Test data. That exact inference with a Gaussian likelihood achieves the strongest LML and weak test set scores implies the ML-II procedure is over-fitting the training data with this likelihood model.

For the Student's t likelihood the performance of each approximate inference method varied significantly. LVB results were uniformly the weakest. We conjecture this is an artifact of the squared

[2] UCI datasets can be downloaded from `archive.ics.uci.edu/ml/datasets/`.

[3] The Friedman dataset is constructed as described in Kuss [2006] Section 5.6.1. and Friedman [1991].

[4] The Neal dataset is constructed as described in Neal [1997] Section 7.

exponential local site bounds employed by the `gpml` toolbox poorly capturing the non log-concave potential functions mass. For Student's t potentials improved LVB performance has been reported by employing bounds that are composed of two terms on disjoint partitions of the domain [Seeger and Nickisch, 2011b], validating their efficacy in the context of Student's t GP regression models is reserved for future work. For the test set metrics G-KL approximate inference achieves the strongest performance.

Broadly, the Laplace likelihood achieved the best results on all datasets. G-EP frequently did not converge for both the Friedman and Concrete Slump Test problems and so results are not presented. Unlike the Student's t likelihood model, results are more consistent across approximate inference methods. G-KL achieves a narrow but consistently superior LML value to LVB. Approximate test set predictive values are roughly the same for all inference methods with LVB achieving a small advantage.

We reiterate that standard G-EP approximate inference, as implemented in the `GPML` toolbox, was used to obtain these results. The authors did not anticipate convergence issues for G-EP in the GP models considered - the Laplace likelihood model's log posterior is concave and the system has full rank. Power G-EP, as proposed in Minka [2004], has previously been shown to have robust convergence for under determined linear models with Laplace potentials [Seeger, 2008]. Similarly, we expect that power G-EP would also exhibit robust convergence in GP models with Laplace likelihoods. Verifying this experimentally and assessing the performance of power G-EP approximate inference in noise robust GP regression models is left for future work.

The G-KL LML uniformly dominates the LVB values. This is theoretically guaranteed for a model with fixed hyperparameters and log-concave site potentials, see Section 4.4.1 and Section 4.2.2. However, the G-KL bound is seen to dominate the local bound even when these conditions are not satisfied. The results show that both G-KL bound optimisation and G-KL hyperparameter optimisation is numerically stable. G-KL approximate inference appears more robust than G-EP and LVB – G-KL hyperparameter optimisation always converged, often to a better local optima.

Summary

The results confirm that the G-KL procedure as a sensible route for approximate inference in GP models with non-conjugate likelihoods. The G-KL procedure is generally applicable in this setting and easy to implement for new likelihood models. Indeed, all that is required to implement G-KL approximate inference for a GP regression model is the pointwise evaluation of the univariate likelihood function $p(y_n|w_n)$. Furthermore, we have seen that G-KL optimisation is numerically robust, in all the experiments G-KL converged and achieved strong performance.

5.2 Bayesian logistic regression : covariance parameterisation

In this section we examine the relative performance, in terms of speed and accuracy of inference, of each of the constrained G-KL covariance decompositions presented in Section 4.3.1.3. As a bench mark, we also compare G-KL approximate inference results to scalable approximate LVB methods with marginal variances approximated using iterative Lanczos methods [Seeger and Nickisch, 2011b]. Our aim is not make a comparison of deterministic approximate inference methods for Bayesian logistic regres-

			$N_{trn}=250$		$N_{trn}=500$		$N_{trn}=2500$	
			$K=25$	$K=50$	$K=25$	$K=50$	$K=25$	$K=50$
Time	G-KL	Chev	0.49±0.02	0.69±0.08	1.25±0.04	1.36±0.04	16.50±0.89	17.31±0.82
		Band	0.96±0.02	1.37±0.02	2.25±0.10	4.06±0.29	24.31±0.96	29.60±1.18
		Sub	0.73±0.01	0.93±0.03	1.41±0.03	1.93±0.04	11.89±0.54	15.26±1.02
		FA	2.05±0.26	2.29±0.21	2.92±0.17	3.47±0.17	20.06±1.51	22.69±2.70
	LVB		0.37±0.00	0.47±0.01	0.46±0.02	0.52±0.00	1.56±0.03	1.85±0.01
$\tilde{\mathcal{B}}$	G-KL	Chev	−1.19±0.01	−1.15±0.01	−0.93±0.01	−0.91±0.01	−0.42±0.00	−0.41±0.00
		Band	−1.15±0.01	−1.09±0.01	−0.92±0.01	−0.88±0.01	−0.42±0.00	−0.41±0.00
		Sub	−3.08±0.02	−2.20±0.01	−1.90±0.01	−1.46±0.01	−0.62±0.00	−0.54±0.00
		FA	−1.19±0.01	−1.17±0.01	−0.93±0.01	−0.91±0.01	−0.41±0.00	−0.40±0.00
	LVB		–±–	–±–	–±–	–±–	–±–	–±–
$\|\mathbf{w}-\mathbf{w}_{tr}\|_2/D$	G-KL	Chev	0.88±0.00	0.87±0.00	0.84±0.00	0.84±0.00	0.64±0.00	0.64±0.00
		Band	0.87±0.00	0.87±0.00	0.84±0.00	0.84±0.00	0.64±0.00	0.64±0.00
		Sub	0.88±0.00	0.87±0.01	0.87±0.00	0.86±0.00	0.71±0.00	0.70±0.00
		FA	0.88±0.00	0.87±0.01	0.84±0.00	0.84±0.00	0.64±0.00	0.64±0.00
	LVB		0.90±0.00	0.89±0.00	0.89±0.00	0.88±0.00	0.72±0.00	0.72±0.00
$\log p(\mathbf{y}^*\|\mathbf{X}^*)/N_{tst}$	G-KL	Chev	−0.58±0.01	−0.58±0.01	−0.50±0.01	−0.49±0.01	−0.18±0.00	−0.18±0.00
		Band	−0.58±0.01	−0.57±0.01	−0.50±0.01	−0.49±0.01	−0.18±0.00	−0.18±0.00
		Sub	−0.72±0.02	−0.65±0.02	−0.63±0.01	−0.59±0.01	−0.20±0.00	−0.20±0.00
		FA	−0.58±0.01	−0.58±0.01	−0.51±0.01	−0.50±0.01	−0.18±0.00	−0.18±0.00
	LVB		−0.75±0.02	−0.77±0.02	−0.63±0.01	−0.64±0.01	−0.20±0.00	−0.20±0.00

Table 5.2: Synthetic Bayesian logistic regression results for a model with unit variance Gaussian prior $\mathbf{w} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ with $\dim(\mathbf{w}) = 500$, likelihood $p(\mathbf{y}|\mathbf{w}, \mathbf{X}) = \prod_{n=1}^{N_{trn}} \sigma(y_n \mathbf{w}^\mathsf{T} \mathbf{x}_n)$, class labels $y_n \in \{+1, -1\}$ and $N_{tst} = 5000$ test points. G-KL results obtained using chevron Cholesky (Chev), banded Cholesky (Band), subspace Cholesky (Sub) and factor analysis (FA) constrained parameterisations of covariance. Results presented are the mean and ± 1 standard error values obtained over the 10 randomly sampled datasets (± 0.00 corresponds to a standard error score less than 0.005). Approximate local variational bounding (LVB) results are obtained using low-rank factorisations of covariance computed using iterative Lanczos methods. The parameter K denotes the size of the constrained covariance parameterisation.

sion models but to investigate the time accuracy trade-offs of each of the constrained G-KL covariance parameterisations.

Given a dataset, $\mathcal{D} = \{(y_n, \mathbf{x}_n)\}_{n=1}^N$ with class labels $y_n \in \{-1, 1\}$ and covariates $\mathbf{x}_n \in \mathbb{R}^D$, Bayesian logistic regression models the class conditional distribution using $p(y = 1|\mathbf{w}, \mathbf{x}) = \sigma\left(\mathbf{w}^\mathsf{T}\mathbf{x}\right)$, with $\sigma(x) := 1/(1 + e^{-x})$ the logistic sigmoid function and $\mathbf{w} \in \mathbb{R}^D$ a vector of parameters. Under a Gaussian prior, $\mathcal{N}(\mathbf{w}|\mathbf{0}, \boldsymbol{\Sigma})$, the posterior is given by

$$p(\mathbf{w}|\mathcal{D}) = \frac{1}{Z}\mathcal{N}(\mathbf{w}|\mathbf{0}, \boldsymbol{\Sigma}) \prod_{n=1}^{N} \sigma\left(y_n \mathbf{w}^\mathsf{T} \mathbf{x}_n\right). \tag{5.2.1}$$

Where we have used the symmetry property of the logistic sigmoid such that $p(y = -1|\mathbf{w}, \mathbf{x}) = 1 - p(y = 1|\mathbf{w}, \mathbf{x}) = \sigma\left(-\mathbf{w}^\mathsf{T}\mathbf{x}\right)$. The expression above is of the form of equation (2.3.1) with log-concave site-projection potentials $\phi_n(x) = \sigma(x)$ and $\mathbf{h}_n = y_n \mathbf{x}_n$.

Experimental setup

We synthetically generate the datasets. The data generating parameter vector $\mathbf{w}_{tr} \in \mathbb{R}^D$ is sampled from a factorising standard normal $\mathbf{w}_{tr} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$. The covariates, $\{\mathbf{x}_n\}_{n=1}^N$, are generated by first sampling an independent standard normal, then linearly transforming these vectors to impose correlation between some of the dimensions, and finally the data is renormalised so that each dimension has unit variance. The covariate linear transformation matrix a sparse square matrix generated as the sum of the identity matrix and a sparse matrix with one element from each row sampled from a standard normal. Class labels $y_n \in \{1, -1\}$ are sampled from the likelihood $p(y_n|\mathbf{w}, \mathbf{x}_n) = \sigma(y_n \mathbf{w}^\mathsf{T} \mathbf{x}_n)$. The inferential model's prior and likelihood distributions are set to match the data generating process.

Results are obtained for a range of dataset dimensions: $D = 250, 500, 1000$ and $N = \frac{1}{2}D, D, 5D$. We also vary the size of the constrained covariance parameterisations, which is reported as K in the result tables. For chevron Cholesky K refers to the number of non-diagonal rows of $\mathbf{C}$. For subspace Cholesky K is the dimensionality of the subspace. For banded Cholesky K refers to the band width of the parameterisation. For the factor analysis (FA) parameterisation K refers to the number of factor loading vectors. For local variational bounding (LVB) approximate inference K refers to the number of Lanczos vectors used to update the variational parameters. The parameter K is varied as a function of the parameter vector dimensionality with $K = 0.05 \times D$ and $K = 0.1 \times D$.

Since the G-KL bound is strongly concave we performed G-KL bound optimisation using Hessian free Newton methods for all the Cholesky parameterised covariance experiments. G-KL bound optimisation was terminated when the largest absolute value of the gradient vector was less than 10^{-3}. For subspace Cholesky we iterated between optimising the subspace parameters $\{\mathbf{m}, \mathbf{C}, c\}$ and updating the subspace basis vectors $\mathbf{E}$ each five times. The subspace vectors were updated using the fixed point iteration with the Lanczos approximation (see Appendix B.2.3 for details). For the FA parameterisation the G-KL bound is not concave so we use LBFGS to perform gradient ascent. All other `minFunc` options were set to default values.

LVB approximate inference is achieved using the `glm-ie 1.4` package [Nickisch, 2012]. LVB inner loop optimisation used nonlinear conjugate gradients with at most 50 iterations. The maximum number of LVB outer loop iterations was set to 10. All other LVB `glm-ie` optimisation settings were set to default values. Results for these experiments were obtained using Matlab 2011a on a Intel E5450 3Ghz machine with 8 cores and 64GB of RAM.

Results

Results for $D = 500$ are presented in Table 5.2. For reasons of space, results for $D = 250$ and $D = 1000$ are presented at the end of this chapter in Table 5.4 and Table 5.5. The tables present average and standard error scores obtained from 10 synthetically generated datasets.

The average convergence time and standard errors of each of the methods is presented in the first row section of the result tables. In the smaller problems considered, the best G-KL times were achieved by the chevron Cholesky covariance followed by the banded, the subspace and the FA parameterisations in that order.

The recorded banded Cholesky convergence times are seen to scale super-linearly with K. This is an implementational artefact. Whilst bound/gradient computations for chevron and banded parameterisations both scale $O(NDK)$ they access and compute different elements of the data and Cholesky matrices. The chevron gradients can be computed using standard matrix multiplications for which Matlab is highly optimised. The banded parameterisation needs to access matrix elements in a manner not standard to Matlab – this implementational issue, despite a Matlab mex C implementation, could not be entirely eliminated.

LVB and chevron G-KL achieved broadly similar convergence times for the $N \leq D$ and $D \leq 500$ experiments with LVB faster in the larger D experiments. LVB is significantly faster than G-KL methods for the $N = 5 \times D$ experiments, possibly this is a consequence of the double-loop structure of the LVB implementation. Whilst the subspace G-KL method is significantly slower in the smaller problems when $D = 1000$ it is the fastest G-KL method, beating LVB in problems where $N \leq D$.

In the result tables, the bound values are normalised by the size of the training set, *i.e.* $\tilde{\mathcal{B}} = \mathcal{B}/N_{trn}$, to make comparisons across models easier. As the training set size increases the normalised bound value increases, presumably reflecting the fact that the posterior tends to a Gaussian in the limit of large data. Furthermore, the difference in bound values between the parameterisations become smaller as the size of the training set increases.

The G-KL banded covariance parameterisation achieves the strongest bound value with the chevron and factor analysis parameterisations a close second place. The subspace bound values are comparatively poor. This is not unexpected since the subspace parameterisation has a single parameter (denoted c in Section 4.3.1.3) that specifies the variance in all directions orthogonal to the subspace vectors $\mathbf{E}$. It is known that the density q that minimises $\mathrm{KL}(q|p)$ tends to seek out the modes of p and avoid those regions of parameter space where p is close to zero. Therefore the parameter c will tend to the smallest value of the variance of p in the directions orthogonal to the subspace vectors, the resulting G-KL bound value will therefore be greatly underestimated. The approximate LVB method does not provide a lower-bound when marginal variances are approximated using low-rank methods and therefore values are not reported in the result tables.

Since these results are obtained from datasets sampled from densities with known parameters we can directly asses the accuracy of the posterior parameter estimate against the ground truth. The posterior mean minimises the expected loss $\|\mathbf{w}_{tr} - \mathbf{w}\|_2$. Thus, in the third row section of the results table, we report the average error $\|\mathbf{w}_{tr} - \mathbf{m}\|_2$ where $\mathbf{m}$ is the mean of the Gaussian posterior approximation $q(\mathbf{w}) = \mathcal{N}(\mathbf{m}, \mathbf{S})$. To make comparisons easier, the ℓ^2 errors are normalised by the dimensionality of the respective models D. The results show that the G-KL mean is broadly invariant to the G-KL covariance parameterisations used. LVB results are noticeably poorer than the G-KL methods.

Approximate test set log predictive probabilities are presented in the fourth row section of the result tables. This metric is arguably the best suited to measure the global accuracy of the posterior approximations since it is an expectation over the support of the approximate posterior [MacKay and Oldfield, 1995]. The values reported in the table are approximated using $\log p(\mathbf{y}^*|\mathbf{X}^*) \approx$

$\sum_n \log \left\langle p(y_n^* \mathbf{w}^\mathsf{T} \mathbf{x}_n^*) \right\rangle_{q(\mathbf{w})}$. The values presented are normalised by the size of the test set where $N_{tst} = 10 \times D$ in all experiments. The results show that chevron, banded and FA parameterisations achieve the best, and broadly similar, performance. Test set predictive accuracy increases for all methods as a function of the training set size. Subspace G-KL and approximate LVB achieve broadly similar and noticeably weaker performance than the other methods.

Summary

The results support the use of the constrained Cholesky covariance parameterisations to drive scalable G-KL approximate inference procedures. Whilst neither the banded nor chevron Cholesky parameterisations are invariant to permutations of the index set they both achieved the strongest bound values and test set performance. Unfortunately, due to implementational issues, the banded Cholesky parameterisation gradients are slow to compute resulting in slower recorded convergence times. The non-concavity of the factor analysis parameterised covariance resulted in slower recorded convergence times than the concave models. Whilst the subspace G-KL parameterisation had poorer performance in the smaller problems it broadly matched or outperformed the approximate LVB method in the largest problems.

5.3 Bayesian logistic regression : larger scale problems

In the previous section we examined the performance of the different constrained parameterisations of G-KL covariance that we proposed in Section 4.3.1 to make G-KL methods fast and scalable. The results presented showed that banded Cholesky, chevron Cholesky and subspace Cholesky factorisations were the generally the most efficient and accurate parameterisations. In this section we apply these constrained covariance G-KL methods and fast approximate local variational bounding (LVB) methods to three large scale real world logistic regression problems.

Experimental Setup

We obtained results for three large scale datasets: `a9a`, `realsim` and `rcv1`.[5] Training and test datasets were randomly partitioned such that: `a9a` $D = 123$, $N_{trn} = 16,000$, $N_{tst} = 16,561$ with the number of non-zero elements in the combined training and test sets (nnz) totalling $nnz = 451,592$; `realsim` $D = 20,958$, $N_{trn} = 36,000$, $N_{tst} = 36,309$ and $nnz = 3,709,083$; `rcv1` $D = 42,736$, $N_{trn} = 50,000$, $N_{tst} = 50,000$ and $nnz = 7,349,450$.

Model parameters were, for the purposes of comparison, fixed to the values stated by Nickisch and Seeger [2009]: τ, a scaling on the likelihood term $p(y_n|\mathbf{w}, \mathbf{x}_n) = \sigma(\tau y_n \mathbf{w}^\mathsf{T} \mathbf{x}_n)$, was set with $\tau = 1$ in the `a9a` dataset and $\tau = 3$ for `realsim` and `rcv1`; the prior covariance was spherical such that $\mathbf{\Sigma} = s^2 \mathbf{I}$ with $s^2 = 1$.

Approximate LVB results were obtained with the `glm-ie` Matlab toolbox using low rank Lanczos factorisations of covariance. The size of the covariance parameterisation is denoted as K in the results table. For the chevron Cholesky parameterisation K refers to the number of non-diagonal rows in the Cholesky matrix. In the subspace Cholesky factorisation K refers to the number of subspace vectors used. For the fast approximate LVB methods K is the number of Lanczos vectors used to approximate

[5] These datasets can be downloaded from `www.csie.ntu.edu.tw/~cjlin/libsvmtools/datasets/`.

covariance. For the a9a dataset $K = 80$, for the other two problems $K = 750$. Approximate LVB optimisation was obtained using five outer loop iterations after which no systematic increase in the LVB approximate bound values was observed.

G-KL results were obtained using the vgai toolbox. Three constrained parameterisations of G-KL covariance were considered: diagonal Cholesky, chevron Cholesky and subspace Cholesky. For the subspace Cholesky results we used the fixed point procedure described in Appendix B.2.3. The G-KL bound was optimised using non-linear conjugate gradients. G-KL bound optimisation was terminated when the largest absolute value of the gradient was < 0.1.

Results

Results are presented in Table 5.3. The LVB bound value for the a9a dataset is calculated by substituting the optimal approximate local variational Gaussian moments into the G-KL bound. The G-KL bound evaluated using the local variational moments cannot be computed in the larger datasets since computing $\mathbf{A}^{-1}$, required to evaluate the G-KL bound, is infeasible. Approximate LVB methods that use low rank approximations of covariance do not provide a lower-bound to the normalisation constant only an approximation to it, thus lower-bound values are not reported for the larger problems in Table 5.3.

The approximate LVB method is significantly faster for the small K a9a problem (albeit with a worse bound) than the G-KL method. In the larger experiments however the results show that G-KL approximate inference utilising constrained Gaussian covariances can achieve similar convergence speeds to fast approximate local bound solvers. The experiments considered here are of significantly larger dimensionality, both in N and D, than the experiments of the previous section. However, both the LVB and G-KL methods achieve fast convergence, this is a consequence of the sparsity of the datasets, evidencing that both algorithms have the desirable property of scaling with respect to the number of non-zero terms in each data vector (the effective dimensionality) and not its raw dimensionality. Test set accuracies are broadly the same for both approximate inference procedures.

Summary

The results presented here provide further evidence of the utility of the constrained parameterisations of covariance described in Section 4.3.1. Using constrained parameterisations of covariance, the G-KL procedure is able to achieve rapid convergence and performance achieving results on a par with state of the art fast approximate local variational bound approximate inference methods.

5.4 Bayesian sparse linear models

Many problems in machine learning and statistics can be addressed by linear models with sparsity inducing prior distributions. Examples include, feature selection in regression problems [Wipf, 2004], source separation [Girolami, 2001], denoising or deblurring problems [Fergus et al., 2006], and signal reconstruction from a set of under-determined observations [Seeger and Nickisch, 2008]. In all of these cases, the prior results in a posteriori parameter estimates that are biased towards sparse solutions. For feature selection problems this assumption can be useful if we believe that only a small subset of the features

	a9a				realsim				rcv1			
	G-KL Full	G-KL Chev	G-KL Sub	LVB	G-KL Diag	G-KL Chev	G-KL Sub	LVB	G-KL Diag	G-KL Chev	G-KL Sub	LVB
K	–	80	80	80	–	100	750	750	–	50	750	750
$\mathcal{B}_{KL}(\mathbf{m}, \mathbf{S})$	$-5,374$	$-5,375$	$-5,379$	$-5,383$	$-5,564$	$-5,551$	$-5,723$	–	$-6,981$	$-6,979$	$-7,286$	–
CPU(s)	85	91	68	5	180	350	575	583	176	424	955	436
Acc. %	15.12	15.10	15.12	15.10	2.86	2.86	2.86	2.87	2.90	2.89	2.94	2.94

Table 5.3: G-KL and approximate local variational bound (LVB) Bayesian binary logistic regression results for the `a9a`, `realsim` and `rcv1` datasets. K refers to the size of the covariance approximation: G-KL chevron Cholesky covariance has K non-diagonal rows; G-KL subspace Cholesky covariance has a K subspace basis vectors and a diagonal Cholesky subspace matrix; approximate local bounding results obtained with K Lanczos vectors. G-KL diag refers to a bandwidth one Cholesky parameterisation of covariance. CPU times were recorded in Matlab R2009a using an Intel 2.5Ghz Core 2 Quad processor.

are necessary to model the data. Using an informative prior is essential in the case of under-determined linear models where there are more sources than signals, in which case hyper-planes in parameter space have equiprobable likelihoods and priors are needed to constrain the space of possible solutions.

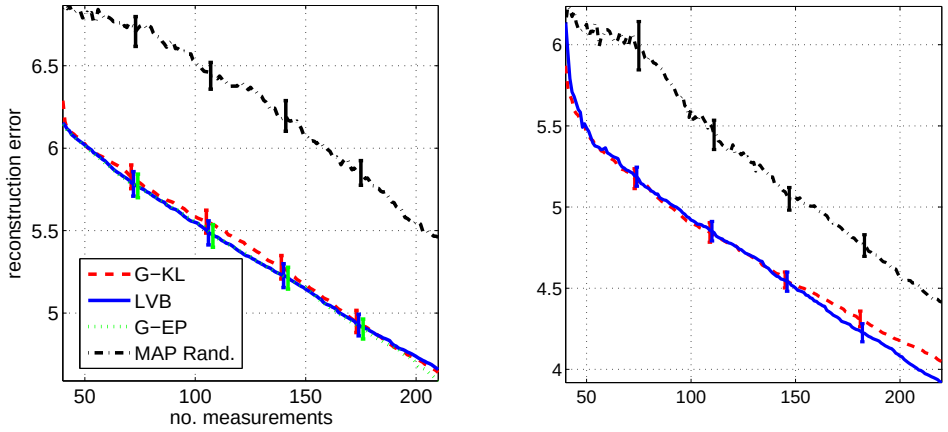
Figure 2.7 depicts the posteriors resulting from an under-determined linear model for a selection of different priors. Since the Laplace prior is log-concave the posterior is unimodal and log-concave. For non log-concave priors the resulting posterior can be multimodal – for instance when $p(\mathbf{w})$ is the Student's t distribution or the sparsity promoting distribution composed from a mixture of Gaussians.

In the case of signal reconstruction, deblurring and source separation sparse priors are used to encode some of the prior knowledge we have about the source signal we wish to recover. Natural images for instance are known to have sparse statistics over a range of linear filters (an example filter being the difference in intensities of neighbouring pixels) [Olshausen and Field, 1996]. Sparse priors that encode this knowledge about the statistics of natural images then bias estimates towards settings that share this statistical similarity.

In this section we consider Bayesian Sequential Experimental Design (SED) for the sparse linear model. At each stage of the SED process we approximate the posterior density of the model parameters and then use the approximate posterior to greedily select new, maximally informative measurements. First, we describe the probabilistic model and the experimental design procedure. Second, we compare approximate inference methods on a small scale artificial SED problem. Third, we compare G-KL and approximate local variational bounding methods for SED on a $64 \times 64 = 4,096$ pixel natural image problem. Our approach broadly follows that laid out by Seeger and Nickisch in [Seeger and Nickisch, 2008, Seeger, 2009, Seeger and Nickisch, 2011b].

Probabilistic model

We observe noisy linear measurements $\mathbf{y} \in \mathbb{R}^N$ assumed to be drawn according to $\mathbf{y} = \mathbf{M}\mathbf{w} + \boldsymbol{\nu}$ where $\mathbf{M} \in \mathbb{R}^{N \times D}$ is the linear measurement matrix with $N \ll D$, $\boldsymbol{\nu} \sim \mathcal{N}\left(\mathbf{0}, \nu^2\mathbf{I}\right)$ is additive Gaussian noise, and $\mathbf{w} \in \mathbb{R}^D$ is the signal that we wish to recover. A sparse prior, here we use either the Laplace or the Student's t, is placed on $\mathbf{s}$ the linear statistics of $\mathbf{w}$ such that $\mathbf{s} = \mathbf{B}\mathbf{w}$. The matrix $\mathbf{B} \in \mathbb{R}^{M \times D}$ is a



(a) Laplace (b) Student's t

Figure 5.2: Sequential experimental design for the Bayesian sparse linear model with synthetic signals. Sparse signals, $\mathbf{w}$, are sampled from (a) a Laplace with $\tau = 0.2$ and (b) a Student's t with $\nu = 3$, $\sigma^2 = 0.0267$.

collection of M linear filters. By placing the prior directly on the statistics, $\mathbf{s}$, the posterior is proportional to the product of the Gaussian likelihood and the sparse prior potentials,

$$p(\mathbf{w}|\mathbf{M}, \mathbf{y}, \boldsymbol{\tau}, \nu^2) \propto \mathcal{N}\left(\mathbf{y}|\mathbf{M}\mathbf{w}, \nu^2\mathbf{I}\right) p(\mathbf{s}), \quad \text{where} \quad \mathbf{s} = \mathbf{B}\mathbf{w}.$$

Since the priors are placed directly on the statistics $\mathbf{s}$ and not $\mathbf{w}$ they are not normalised densities with respect to $\mathbf{w}$, as a consequence $\mathcal{B}_{G\text{-}KL}(\mathbf{m}, \mathbf{S})$ is no longer a lower-bound to $\log Z$. However, since the normalisation constant of $p(\mathbf{s})$ is constant with respect to $\mathbf{w}$ ignoring this constant does not affect the G-KL approximation to the posterior density.

Sequential experimental design

SED for the sparse linear model described above is the problem of iteratively choosing which new measurement vectors, $\mathbf{M}^*$, to append to $\mathbf{M}$ so as to maximise subsequent estimation accuracy. Bayesian SED iterates between estimating the posterior density on $\mathbf{w}$, conditioned on current observations, and then using this density to select which new measurements to make. Following Seeger and Nickisch [2011b] we use the information gain metric to decide which measurement vectors will be maximally informative. Information gain is defined as the difference in differential Shannon information of the posterior density before and after the inclusion of new measurements and their corresponding observations. For the linear model we consider, it is given by

$$I_{gain}(\mathbf{M}^*) = H\left[p(\mathbf{w}|\mathbf{M}, \mathbf{y})\right] - H\left[p(\mathbf{w}|\mathbf{M}, \mathbf{y}, \mathbf{M}^*, \mathbf{y}^*)\right], \tag{5.4.1}$$

where $H\left[p(x)\right] := -\left\langle \log p(x) \right\rangle_{p(x)}$ is the Shannon differential entropy. Since inference is not analytically tractable we cannot access either of the densities required by equation (5.4.1). We can, however, obtain an approximation to the information gain by substituting in a Gaussian approximation to the posterior. Doing so with $p(\mathbf{w}|\mathbf{M}, \mathbf{y}) \approx \mathcal{N}(\mathbf{w}|\mathbf{m}, \mathbf{S})$ we have $\left\langle \log p(\mathbf{w}|\mathbf{M}, \mathbf{y}) \right\rangle \approx \frac{1}{2} \log\det(\mathbf{S}) + c$ with c

an additive constant. The second entropy is obtained by Gaussian conditioning on the joint approximate Gaussian density $p(\mathbf{w}, \mathbf{y}^*|\mathbf{y}) \propto \mathcal{N}(\mathbf{w}|\mathbf{m}, \mathbf{S})\mathcal{N}\left(\mathbf{y}^*|\mathbf{M}^*\mathbf{w}, \nu^2\mathbf{I}\right)$. The approximation to the information gain can then be written as

$$I_{gain}(\mathbf{M}^*) \approx \frac{1}{2}\log\det\left(\mathbf{M}^*\mathbf{S}\mathbf{M}^{*\mathsf{T}} + \nu^2\mathbf{I}\right) + c. \tag{5.4.2}$$

If we constrain the measurements to have unit norm $I_{gain}(\mathbf{M}^*)$ above will be maximised when the rows of $\mathbf{M}^*$ lie along the leading principal eigenvectors of the approximate posterior covariance $\mathbf{S}$. These eigenvectors are approximated in our experiments using iterative Lanczos methods.

Synthetic signals

Initially we consider applying sequential experimental design to a sparse signal reconstruction problem using small scale synthetic signals. In this artificial set up we wish to recover some signal $\mathbf{w}_{tr} \in \mathbb{R}^{512}$ from a set of noisy linear measurement $\mathbf{y} \in \mathbb{R}^m$ where $m \ll 512$. We initialised the experiments with $m_0 = 40$ random unit norm linear measurement vectors $\mathbf{M} \in \mathbb{R}^{m_0 \times 512}$.

In this setup we placed the sparse prior directly on $\mathbf{w}$ with $\mathbf{B} = \mathbf{I}$. Sparse signals, $\mathbf{w}_{tr}$, were sampled independently over dimensions from either the Laplace ($\mu = 0, \tau = 0.2$) or the Student's t ($\nu = 3, \sigma^2 = 0.027$) densities. Noisy linear measurements were sampled from the source signals with $\mathbf{y} \sim \mathcal{N}\left(\mathbf{M}\mathbf{w}_{tr}, \nu^2\mathbf{I}\right)$ and $\nu^2 = 0.005$ throughout. Model priors and likelihoods were fixed to match the data generating densities.

For the Laplace generated signals we applied G-KL, local variational bounding (LVB) and power G-EP ($\eta = 0.9$) approximate inference methods. G-EP and LVB results were obtained using the publicly available `glm-ie` Matlab toolbox. Since the model is of sufficiently small dimensionality approximate covariance decompositions were not required. For the Student's t generated signals only G-KL and LVB approximate inference methods were applied since G-EP is unstable in this setting.

For the Laplace signals, when $D = 512$ and $N = 110$, inference takes 0.3 seconds for LVB, 0.6 seconds for G-EP, and 1.6 seconds for G-KL.[6] For the Student's t signals, again with $D = 512$ and $N = 110$, inference takes 0.3 seconds for LVB and 6 seconds for G-KL. For Laplace signals, for which the G-KL bound is concave, gradient ascent was performed using a Hessian free Newton method with finite differences approximation for Hessian vector products (see Chapter 7 Nocedal and Wright [2006]). For the Student's t signals, for which the G-KL bound is not guaranteed to be concave or even unimodal, gradient ascent was performed using nonlinear scaled conjugate gradients. G-KL optimisation was terminated in both settings once the largest absolute value of the bound's gradient was less than 0.01. LVB and G-EP were optimised for seven outer loop iterations after which no systematic improvement in the approximate $\log Z$ value was observed.

ℓ^2 norm reconstruction error mean and standard error scores obtained over the 25 experiments conducted are presented in Figure 5.2. For the Laplace generated signals LVB, G-EP and G-KL approximate inference procedures provide broadly the same reconstruction error performance. All sequentially designed procedures outperform MAP estimates with standard normal random measurements. The im-

[6] Experiments were timed using MATLAB R2009a on a 32 bit Intel Core 2 Quad 2.5 GHz processor.

proved performance comes mainly in the first few iterations of the SED process with all methods achieving broadly similar iterative improvements in reconstruction error after that. For the Student's t prior again LVB and G-KL procedures obtain broadly the same performance with G-KL appearing to become slightly less effective towards the end of the experiment.

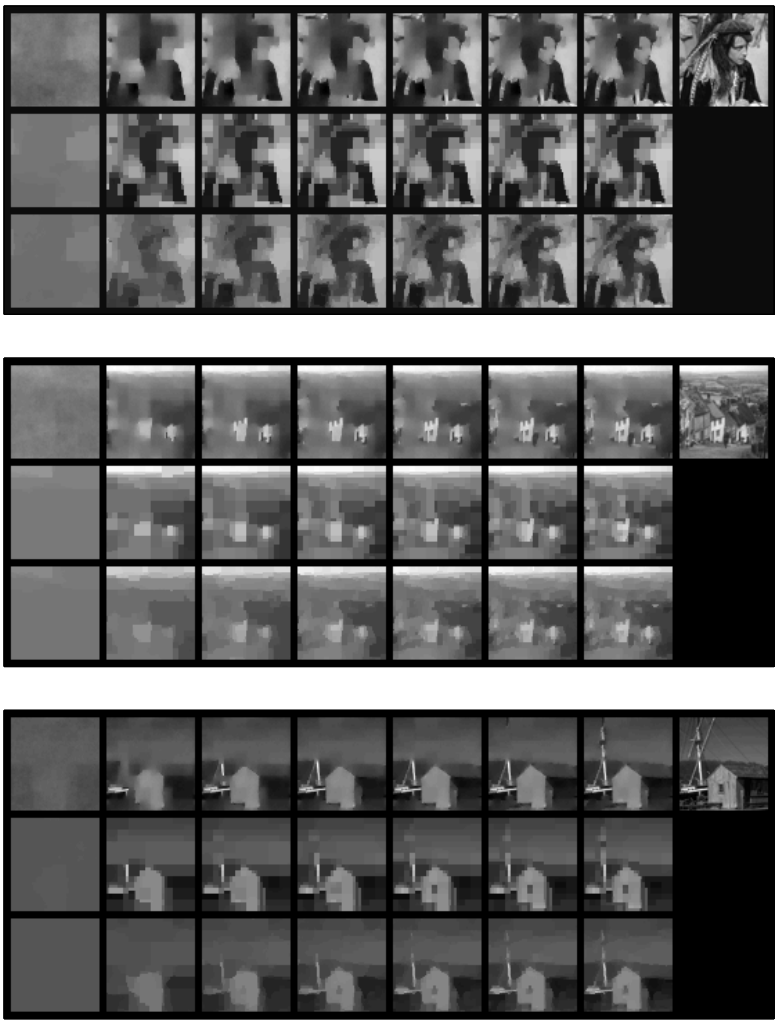


Figure 5.3: Reconstructed images from the Bayesian sequential experimental design (SED) experiments. We plot the estimated images obtained by each approximate inference procedure at different stages of the SED process. Each pane corresponds to a different underlying image. The true image is shown in the last image of the first row of each pane. Otherwise, the first row of each pane plots the G-KL mean, the second row the LVB mean and the third row the MAP reconstruction with randomly selected measurement vectors. The k^{th} column of each pane plots the estimated image using $100 + 300 \times (k-1)$ measurements.

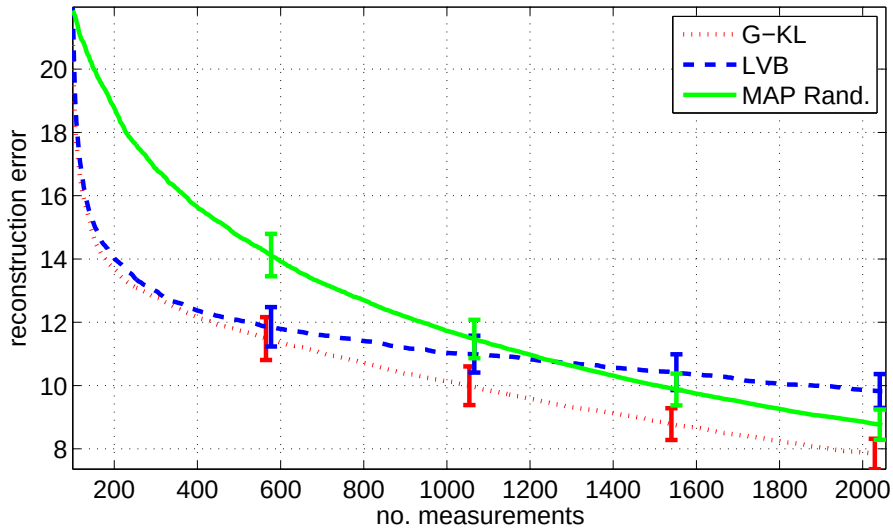


Figure 5.4: ℓ^2 reconstruction errors for the natural image sequential experimental design task. Mean and standard error scores are presented averaged over 16 different 64×64 pixel images.

Natural images

We consider sequential experimental design for the problem of recovering natural images from a set of under-determined noisy linear measurements. This problem is modelled by placing priors on the statistics of natural images that are known to exhibit sparsity. These statistics can be captured by suitable linear projections of the image vector (formed by concatenating the pixel value columns of the image). For the results presented we employ two types of image filter known to exhibit sparse statistics in natural images: finite differences, the difference in intensity values of horizontally or vertically neighbouring pixels; and multi-scale orthonormal wavelet transforms, constructed using the Daubechies four wavelet (see Seeger and Nickisch [2011b] for further details). Both filters can be expressed as extremely sparse vectors, the set of which is collected in the matrix $\mathbf{B}$, giving $\mathbf{B} \in \mathbb{R}^{M \times D}$ where $M = 3 \times D$. Image filters were implemented using the `glm-ie` package. Laplace priors placed on each of the linear filter responses had $\tau = 0.1$ for the finite difference filters and $\tau = 0.14$ for the wavelet filters. This experimental approach follows that laid out in Chapter 5 of Nickisch [2010].

We apply the SED procedure detailed above by iteratively approximating the posterior density $p(\mathbf{w}|\mathbf{y}, \mathbf{M}, \mathbf{B}, \boldsymbol{\tau}, \nu^2)$ where: $\mathbf{w} \in \mathbb{R}^D$ corresponds to the unknown image vector; $\mathbf{y} \in \mathbb{R}^N$ the noisy measurements where $N \ll D$; and $\mathbf{M} \in \mathbb{R}^{N \times D}$ is the linear measurement matrix constrained to have rows with unit norm. The measurement matrix is initialised with 100 standard normal randomly sampled vectors normalised to have unit norm. The sequential experimental design process approximates the posterior based on current measurements and the prior, these are then used to select new unit norm linear measurement vectors $\mathbf{M}^* \in \mathbb{R}^{3 \times D}$ to append to $\mathbf{M}$. New observations are then synthetically generated by drawing samples from the Gaussian $\mathbf{y}^* \sim \mathcal{N}\left(\mathbf{M}\mathbf{w}_{tr}, \nu^2\mathbf{I}\right)$. In the experiments conducted we use $64 \times 64 = 4096 = D$ pixel grey scale images. The images were down sampled from a collection

frequently used by the vision community,[7] gray scale pixel intensities were linearly transformed to lie in $[-1, 1]$. The likelihood model was fixed with $\nu^2 = 0.005$.

In this larger setting we apply G-KL and LVB approximate inference methods only and make use of approximate covariance decompositions. For G-KL approximate inference we use the chevron Cholesky decomposition with 80 non-diagonal rows. The chevron Cholesky parameterisation was chosen due to its strong performance in previous experiments with respect to both convergence time and accuracy of inference – see Section 5.2. LVB inference is applied with low rank decompositions of covariance using 80 Lanczos vectors. For the first iteration of the SED procedure, $N_0 = 100$, G-KL converged in 30 seconds and LVB in 5 seconds. At each iteration of the SED process each inference procedure was initialised with the posterior from the previous SED iteration. When $N = 2048$ updating the Gaussian approximate posterior took 60 seconds for G-KL and 25 seconds for LVB. Convergence of LVB inference is difficult to asses since the double loop algorithm with Lanczos approximated covariance is not guaranteed at each iteration to increase the approximated marginal likelihood. We iterated the LVB procedure for seven outer loop iterations at which point no systematic increases of approximate marginal likelihood values were observed. Fluctuations in LVB approximate marginal likelihood value in subsequent iterations were roughly ± 10. G-KL inference was terminated when the greatest absolute value of the bounds gradient was less than 0.1, at which point G-KL bound values increased by less than 0.5 per iteration. These results highlight a general distinction between the two methods, LVB optimisation is an approximate EM algorithm whilst G-KL optimisation in this setting is implemented using an approximate second order gradient ascent procedure. EM is often reported to exhibit rapid convergence to low accuracy solutions but can be very slow at achieving high accuracy solutions [Salakhutdinov et al., 2003].

Reconstruction error results are plotted in Figure 5.4. We can see that SED offers greater reconstruction accuracy over random designs for a fixed budget of measurements. Up to roughly 400 designed measurement vectors both G-KL and LVB procedures achieve similar reconstruction errors, after which the rate of LVB iterative performance slows down eventually being overtaken by MAP reconstruction without design (MAP Rand). The reasons for this phenomenon are unclear. As more measurements are added the posterior density will become more spherical, for approximately spherical posteriors the benefit of design over simply adding random measurements is negligible. This could possibly explain the observation that G-KL and the MAP Rand procedures have similar gradients in Figure 5.4 towards the end of the experiment. Why the performance of LVB approximate inference in particular degrades as more observations are added is not clear. One possible explanation is due to the Lanczos covariance approximation, as the posterior becomes increasingly spherical its spectrum will get flatter and the low-rank approximate factorisation may cause degraded Gaussian mean estimation.

Figure 5.3 displays the estimated deconvolved images at different stages of the SED process. Specifically we plot the G-KL and LVB Gaussian mean estimates and the randomly designed MAP estimate. Interestingly, each method displays different visual traits with regards to the quality of the reconstructed

[7] Images were downloaded from `decsai.ugr.es/cvg/dbimagenes/index.php`.

image. G-KL estimates have patches with high fidelity and patches with low fidelity and a soft cloudy texture. LVB and MAP Rand estimates appear more pixelated than the G-KL estimates with image accuracy more uniform across the image pane.

5.5 Summary

The results presented in this chapter confirm that the G-KL approach to approximate inference, as proposed in Chapter 4, is a widely applicable, accurate, fast and scalable deterministic approximate inference method in latent linear models.

One of the principal advantages of the G-KL procedure is the ease with which it can be implemented and applied to new latent linear models. G-KL approximate inference places few restrictions on the model's potential functions $\{\phi_n\}$ to which it can be applied. In Section 5.1 and Section 5.4 we saw that the Gaussian expectation propagation and Laplace approximations could be impractical due to the potentials being either non-differentiable or not log-concave. Whilst local variational bounding methods could be applied to the Student's t Gaussian process regression model the site potential bounds resulted in poor performance. Whereas the G-KL procedure was applicable to each of these models, did not require deriving complicated updates or novel site bounds and achieved strong results and robust convergence in each setting.

The results also confirm that G-KL approximate inference is comparatively accurate versus other deterministic Gaussian approximate inference methods. In each experimental domain considered the G-KL procedure achieved competitive accuracies versus the other Gaussian approximate inference methods considered. What is more the G-KL approximation returns a rigorous lower-bound on the marginal likelihood using either full or constrained parameterisations of covariance. As was expected, using a full Cholesky covariance in the GP experiments, the G-KL bound uniformly dominated the LVB bound.

In Sections 5.2,5.3 and 5.4 we saw that using constrained parameterisations of covariance the G-KL method could be made scalable and fast. Fast approximate solvers for local variational bounding methods are one of the most scalable global, non-factorising, Gaussian approximate inference methods in latent variable models. The results presented show that G-KL approximate inference, with constrained covariance parameterisations, and off the shelf gradient based optimisation methods can achieve comparable convergence times and performance.

5.6 Bayesian logistic regression result tables

			$N_{trn} = 125$		$N_{trn} = 250$		$N_{trn} = 1250$	
			$K = 13$	$K = 25$	$K = 13$	$K = 25$	$K = 13$	$K = 25$
Time	G-KL	Chev	0.14±0.01	0.16±0.00	0.32±0.02	0.34±0.01	3.31±0.09	3.38±0.14
		Band	0.21±0.01	0.28±0.01	0.41±0.01	0.53±0.01	4.05±0.09	4.64±0.09
		Sub	0.42±0.05	0.46±0.02	0.69±0.03	0.81±0.04	4.24±0.15	5.17±0.28
		FA	0.75±0.05	0.74±0.05	0.94±0.08	1.12±0.08	6.18±0.61	5.49±0.40
	LVB		0.27±0.01	0.28±0.00	0.29±0.00	0.31±0.01	0.46±0.01	0.45±0.00
$\tilde{\mathcal{B}}$	G-KL	Chev	−1.08±0.02	−1.05±0.02	−0.89±0.01	−0.87±0.01	−0.41±0.00	−0.40±0.00
		Band	−1.05±0.02	−1.00±0.01	−0.88±0.01	−0.85±0.01	−0.41±0.00	−0.40±0.00
		Sub	−2.93±0.01	−2.11±0.02	−1.83±0.01	−1.43±0.01	−0.60±0.00	−0.52±0.00
		FA	−1.08±0.02	−1.06±0.02	−0.89±0.01	−0.87±0.01	−0.40±0.00	−0.39±0.00
	LVB		–±–	–±–	–±–	–±–	–±–	–±–
$\|\mathbf{m} - \mathbf{w}_{tr}\|_2/D$	G-KL	Chev	1.48±0.01	1.48±0.01	1.38±0.01	1.38±0.01	1.11±0.01	1.11±0.01
		Band	1.48±0.01	1.48±0.01	1.38±0.01	1.38±0.01	1.11±0.01	1.11±0.01
		Sub	1.49±0.01	1.48±0.01	1.43±0.01	1.41±0.01	1.20±0.01	1.18±0.01
		FA	1.48±0.01	1.48±0.01	1.38±0.01	1.38±0.01	1.11±0.01	1.11±0.01
	LVB		1.52±0.01	1.51±0.01	1.45±0.01	1.45±0.02	1.21±0.01	1.21±0.01
$\log p(\mathbf{y}^*\|\mathbf{X}^*)/N_{tst}$	G-KL	Chev	−0.57±0.01	−0.56±0.01	−0.47±0.01	−0.47±0.01	−0.19±0.00	−0.19±0.00
		Band	−0.56±0.01	−0.56±0.01	−0.47±0.01	−0.46±0.01	−0.19±0.00	−0.19±0.00
		Sub	−0.67±0.02	−0.63±0.02	−0.57±0.02	−0.54±0.02	−0.21±0.01	−0.20±0.01
		FA	−0.57±0.01	−0.57±0.01	−0.48±0.01	−0.47±0.01	−0.19±0.00	−0.19±0.00
	LVB		−0.68±0.02	−0.68±0.02	−0.57±0.01	−0.56±0.01	−0.21±0.01	−0.21±0.01

Table 5.4: Bayesian logistic regression covariance parameterisation comparison results for a unit variance Gaussian prior, with parameter dimension $D = 250$ and number of test points $N_{tst} = 2500$. Experimental setup and metrics are described in Section 5.2

			$N_{trn} = 500$		$N_{trn} = 1000$		$N_{trn} = 5000$	
			$K = 50$	$K = 100$	$K = 50$	$K = 100$	$K = 50$	$K = 100$
Time	G-KL	Chev	2.68±0.04	3.37±0.05	6.41±0.11	7.28±0.14	75.23±1.51	78.38±2.10
		Band	6.66±0.58	8.97±0.14	12.81±0.15	20.59±0.26	127.69±2.36	190.65±3.47
		Sub	1.59±0.07	2.58±0.12	3.24±0.03	7.71±0.20	56.67±1.80	75.35±1.63
		FA	9.94±1.00	12.24±0.63	16.21±0.74	18.64±1.09	70.87±3.92	82.13±5.83
	LVB		1.78±0.03	2.65±0.05	4.12±0.04	6.17±0.07	21.88±0.03	33.87±0.02
$\tilde{\mathcal{B}}$	G-KL	Chev	−1.28±0.01	−1.24±0.01	−0.99±0.00	−0.96±0.00	−0.42±0.00	−0.41±0.00
		Band	−1.24±0.01	−1.17±0.01	−0.98±0.00	−0.94±0.00	−0.42±0.00	−0.42±0.00
		Sub	−5.40±0.23	−4.54±0.25	−7.56±0.00	−1.52±0.00	−0.62±0.00	−0.54±0.00
		FA	−1.29±0.01	−1.26±0.01	−1.00±0.00	−0.97±0.00	−0.42±0.00	−0.41±0.00
	LVB		–±–	–±–	–±–	–±–	–±–	–±–
$\|\mathbf{w} - \mathbf{w}_{tr}\|_2/D$	G-KL	Chev	0.53±0.00	0.53±0.00	0.49±0.00	0.49±0.00	0.38±0.00	0.38±0.00
		Band	0.53±0.00	0.53±0.00	0.49±0.00	0.49±0.00	0.38±0.00	0.38±0.00
		Sub	0.56±0.00	0.55±0.00	0.56±0.00	0.50±0.00	0.44±0.00	0.43±0.00
		FA	0.53±0.00	0.53±0.00	0.49±0.00	0.49±0.00	0.38±0.00	0.38±0.00
	LVB		0.54±0.00	0.54±0.00	0.52±0.00	0.52±0.00	0.45±0.00	0.45±0.00
$\log p(\mathbf{y}^*\|\mathbf{X}^*)/N_{tst}$	G-KL	Chev	−0.62±0.01	−0.61±0.01	−0.51±0.01	−0.49±0.01	−0.18±0.00	−0.18±0.00
		Band	−0.61±0.01	−0.59±0.01	−0.50±0.01	−0.49±0.01	−0.18±0.00	−0.18±0.00
		Sub	−0.62±0.01	−0.61±0.01	−0.69±0.00	−0.61±0.01	−0.21±0.00	−0.21±0.00
		FA	−0.62±0.01	−0.61±0.01	−0.52±0.01	−0.51±0.01	−0.18±0.00	−0.18±0.00
	LVB		−0.88±0.01	−0.95±0.02	−0.68±0.01	−0.70±0.01	−0.21±0.00	−0.21±0.00

Table 5.5: Bayesian logistic regression results for a unit variance Gaussian prior, with parameter dimension $D = 1000$ and number of test points $N_{tst} = 5000$. Experimental setup and metrics are described in Section 5.2.

Chapter 6

Affine independent KL approximate inference

In Chapters 4 and 5 we showed that using a Gaussian approximating density and the KL variational objective we could achieve comparatively accurate and efficient approximate inferences versus other deterministic Gaussian approximate inference methods. It is therefore an important avenue of research to develop KL bounding methods further by extending the class of tractable approximating distributions beyond the multivariate Gaussian. Whilst simple mixtures of Gaussians have previously been developed these typically require additional bounds to compute the entropy and can result in computationally demanding optimisation problems [Bishop et al., 1998, Gershman et al., 2012, Bouchard and Zoeter, 2009].

In this chapter we present a procedure to evaluate and optimise the KL bound over a flexible class of approximating variational densities that includes the multivariate Gaussian as a special case. Specifically, we consider optimising the KL bound for variational 'affine independent' densities $q(\mathbf{w})$ constructed as an affine transformation of an independently distributed latent density $q(\mathbf{v})$. In Section 6.2 we introduce and discuss the affine independent density class. In Section 6.3 and 6.4 we show how the KL bound can be evaluated and optimised over this density class. In Section 6.5 we discuss some of the numerical issues associated with the proposed method. In Section 6.7 we present results showing the efficacy of this procedure. Finally, in Section 6.9 we discuss directions for future work.

6.1 Introduction

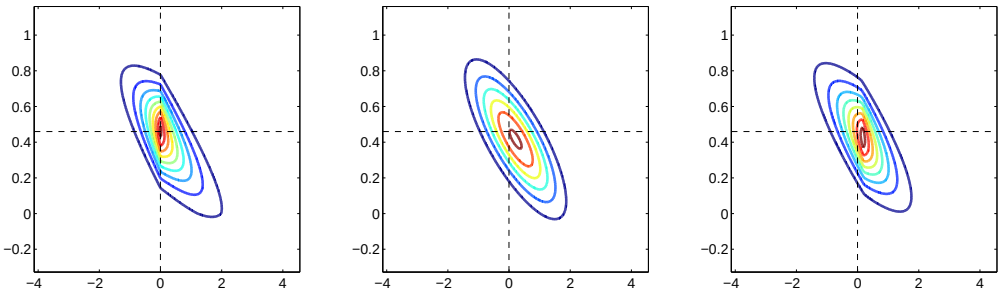
Similar to previous chapters, we seek to perform approximate inference in the latent linear model class. Specifically, for a vector of parameters $\mathbf{w} \in \mathbb{R}^D$, a multivariate Gaussian density $\mathcal{N}(\mathbf{w}|\boldsymbol{\mu}, \boldsymbol{\Sigma})$, we seek to approximate the density defined as

$$p(\mathbf{w}) = \frac{1}{Z}\mathcal{N}(\mathbf{w}|\boldsymbol{\mu}, \boldsymbol{\Sigma}) \prod_{n=1}^{N} \phi_n(\mathbf{w}^\mathsf{T}\mathbf{h}_n), \tag{6.1.1}$$

and its normalisation constant Z defined as

$$Z = \int \mathcal{N}(\mathbf{w}|\boldsymbol{\mu}, \boldsymbol{\Sigma}) \prod_{n=1}^{N} \phi_n(\mathbf{w}^\mathsf{T}\mathbf{h}_n) d\mathbf{w}, \tag{6.1.2}$$

where $\phi_n : \mathbb{R} \to \mathbb{R}^+$ are non-Gaussian, real valued, positive potential functions and $\mathbf{h}_n \in \mathbb{R}^D$ are fixed real valued vectors. Note, the inference problem defined above is equivalent to that specified in Section 2.3, we reproduce it here only for clarity of exposition.



(a) (b) (c)

Figure 6.1: Two dimensional Bayesian sparse linear regression posterior specified by a Laplace prior $\phi_d(\mathbf{w}) \equiv \frac{1}{2\tau}e^{-|w_d|/\tau}$ with $\tau = 0.16$ and Gaussian likelihood $\mathcal{N}\left(y|\mathbf{w}^\mathsf{T}\mathbf{h}, \sigma_l^2\right)$, $\sigma_l^2 = 0.05$ and two data points $\mathbf{h}, y$. (a) True posterior with $\log Z = -1.4026$. (b) Optimal Gaussian approximation with bound value $\mathcal{B}_G = -1.4399$. (c) Optimal AI generalised-normal approximation with bound value $\mathcal{B}_{AI} = -1.4026$.

We approach the problem of forming an approximation $q(\mathbf{w})$ to $p(\mathbf{w})$ and a lower-bound to $\log Z$ using the $\mathrm{KL}(q(\mathbf{w})|p(\mathbf{w}))$ divergence as a variational objective function. As described in Section 3.2, the KL divergence $\mathrm{KL}(q(\mathbf{w})|p(\mathbf{w}))$ provides a lower-bound on $\log Z$ in the form

$$\log Z \geq \mathcal{B}_{KL} := H[q(\mathbf{w})] + \langle \log \mathcal{N}(\mathbf{w}|\boldsymbol{\mu}, \boldsymbol{\Sigma})\rangle + \sum_{n=1}^{N} \left\langle \log \phi_n(\mathbf{w}^\mathsf{T}\mathbf{h}_n) \right\rangle,$$

where the expectations $\langle\cdot\rangle$ are taken with respect to the variational density $q(\mathbf{w})$. Optimising the lower-bound $\mathcal{B}_{KL}$ with respect the density $q(\mathbf{w})$ we can find the 'tightest' lower-bound to $\log Z$ and the 'closest' approximation to $p(\mathbf{w})$. The larger the set of approximating distributions $q(\mathbf{w})$ that this optimisation can be performed over the more accurate this approximate inference procedure has the potential to be. In Chapter 4 we considered multivariate Gaussian $q(\mathbf{w})$ approximations. In this chapter we introduce a more flexible class of approximating densities which we call the affine independent density class and show how the KL bound can be efficiently evaluated and optimised with respect to it.

6.2 Affine independent densities

We first consider independently distributed latent variables $\mathbf{v} \sim q_\mathbf{v}(\mathbf{v}|\boldsymbol{\theta}) = \prod_{d=1}^{D} q_{v_d}(v_d|\theta_d)$ with 'base' distributions q_{v_d}. To enrich the representation, we form the affine transformation $\mathbf{w} = \mathbf{A}\mathbf{v} + \mathbf{b}$ where $\mathbf{A} \in \mathbb{R}^{D\times D}$ is invertible and $\mathbf{b} \in \mathbb{R}^D$. The distribution on $\mathbf{w}$ is then[1]

$$q_\mathbf{w}(\mathbf{w}|\mathbf{A}, \mathbf{b}, \boldsymbol{\theta}) = \int \delta(\mathbf{w} - \mathbf{A}\mathbf{v} - \mathbf{b})\, q_\mathbf{v}(\mathbf{v}|\boldsymbol{\theta}) d\mathbf{v} = \frac{1}{|\det(\mathbf{A})|} \prod_d q_{v_d}\left(\left[\mathbf{A}^{-1}(\mathbf{w} - \mathbf{b})\right]_d |\theta_d\right) \quad (6.2.1)$$

where $\delta(\mathbf{h}) = \prod_d \delta(h_d)$ is the Dirac delta function, $\boldsymbol{\theta}^\mathsf{T} = [\theta_1, ..., \theta_d]$ and $[\mathbf{h}]_d$ refers to the d^{th} element of the vector $\mathbf{h}$. Typically we assume the base distributions are homogeneous, $q_{v_d} \equiv q_v$. For instance, if we

[1] This construction is equivalent to a form of square noiseless independent components analysis. See Ferreira and Steel [2007] and Sahu et al. [2003] for similar constructions.

constrain each factor $q_{v_d}(v_d|\theta_d)$ to be the standard normal $\mathcal{N}(v_d|0,1)$ then $q_{\mathbf{w}}(\mathbf{w}) = \mathcal{N}\left(\mathbf{w}|\mathbf{b}, \mathbf{A}\mathbf{A}^\mathsf{T}\right)$. By using, for example, Student's t, Laplace, logistic, generalised-normal or skew-normal base distributions, equation (6.2.1) parameterises multivariate extensions of these univariate distributions. This class of multivariate distributions has the important property that, unlike the Gaussian, they can approximate skew and/or heavy-tailed $p(\mathbf{w})$. See figures 6.1, 6.2 and 6.3, for examples of two dimensional distributions $q_{\mathbf{w}}(\mathbf{w}|\mathbf{A}, \mathbf{b}, \boldsymbol{\theta})$ with skew-normal and generalised-normal base distributions used to approximate toy machine learning problems.

Provided we choose a base distribution class that includes the Gaussian as a special case (for example generalised-normal, skew-normal and asymptotically Student's t) we are guaranteed to perform at least as well as classical multivariate Gaussian KL approximate inference.

Choosing a dimensionally homogenous base density, that is $q_{v_d} \equiv q_v$ for all d, we note that we may arbitrarily permute the indices of $\mathbf{v}$. Furthermore, since every invertible matrix is expressible as $\mathbf{LUP}$ for $\mathbf{L}$ lower, $\mathbf{U}$ upper and $\mathbf{P}$ permutation matrices, without loss of generality, we may use an LU decomposition to parameterise $\mathbf{A}$ such that $\mathbf{A} = \mathbf{LU}$. Doing so, therefore, incurs no loss in expressibility of $q(\mathbf{w})$ whilst reducing the complexity of subsequent computations.

Whilst defining such affine independent (AI) distributions is straightforward, critically we require that the KL bound, equation (3.2.4), is fast to compute. As we explain below, this can be achieved using the Fourier transform both for the bound and its gradients. Full derivations of these results are presented in Appendix C.

6.3 Evaluating the AI-KL bound

The KL bound can be readily decomposed as

$$\mathcal{B}_{KL} = \underbrace{\log|\det(\mathbf{A})| + \sum_{d=1}^{D} H\left[q(v_d|\theta_d)\right]}_{\text{Entropy}} + \underbrace{\left\langle \log \mathcal{N}(\mathbf{w}|\boldsymbol{\mu}, \boldsymbol{\Sigma})\right\rangle + \sum_{n=1}^{N}\left\langle \log \phi_n(\mathbf{w}^\mathsf{T}\mathbf{h}_n)\right\rangle}_{\text{Energy}}, \tag{6.3.1}$$

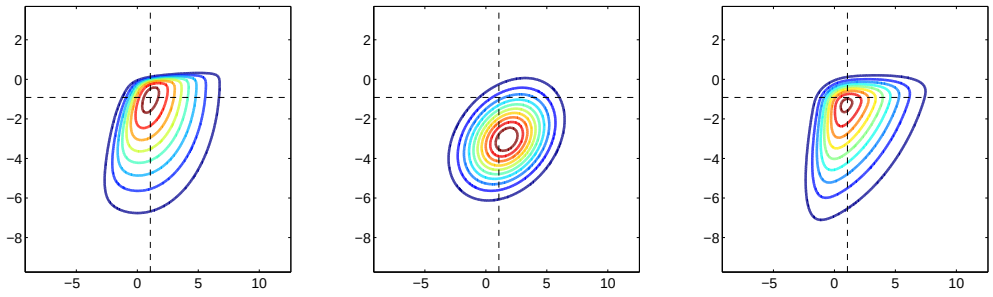
where we used $H\left[q_{\mathbf{w}}(\mathbf{w})\right] = \log|\det(\mathbf{A})| + \sum_d H\left[q_{v_d}(v_d|\theta_d)\right]$ – see for example Cover and Thomas [1991]. For many standard base distributions the entropy $H\left[q_{v_d}(v_d|\theta_d)\right]$ is closed form. When the entropy of a univariate base distribution is not analytically available, we assume it can be cheaply evaluated numerically. The energy contribution to the KL bound is the sum of the expectation of the log Gaussian term, which requires only first and second order moments, and the nonlinear 'site-projections'. The non-linear site-projections, and their gradients, can be evaluated using the methods described below.

6.3.1 Site-projection potentials

Defining $y := \mathbf{w}^\mathsf{T}\mathbf{h}$, the expectation of the site-projection function $\psi : \mathbb{R} \to \mathbb{R}$ and fixed vector $\mathbf{h}$ is equivalent to a one-dimensional expectation, $\left\langle \psi\left(\mathbf{w}^\mathsf{T}\mathbf{h}\right)\right\rangle_{q_{\mathbf{w}}(\mathbf{w})} = \left\langle \psi(y)\right\rangle_{q_y(y)}$ with

$$q_y(y) = \int \delta(y - \mathbf{h}^\mathsf{T}\mathbf{w}) q_{\mathbf{w}}(\mathbf{w}) d\mathbf{w} = \int \delta(y - \boldsymbol{\alpha}^\mathsf{T}\mathbf{v} - \beta) q_{\mathbf{v}}(\mathbf{v}) d\mathbf{v},$$

where $\mathbf{w} = \mathbf{A}\mathbf{v} + \mathbf{b}$ and $\boldsymbol{\alpha} := \mathbf{A}^\mathsf{T}\mathbf{h}$, $\beta := \mathbf{b}^\mathsf{T}\mathbf{h}$. If $\mathbf{h} = \mathbf{e}_d$ with $\mathbf{e}_d$ the d^{th} standard normal basis vector the equation above defines the axis aligned marginal $q(y) = q(w_d|\mathbf{A}, \mathbf{b}, \boldsymbol{\theta})$. We can rewrite this



(a) (b) (c)

Figure 6.2: Two dimensional Bayesian logistic regression posterior defined by the Gaussian prior $\mathcal{N}(\mathbf{w}|\mathbf{0}, 10\mathbf{I})$ and the logistic sigmoid likelihood $\phi_n(\mathbf{w}) = \sigma(\tau_l c_n \mathbf{w}^\mathsf{T}\mathbf{h}_n)$, $\tau_l = 5$. Here $\sigma(x)$ is the logistic sigmoid and $c_n \in \{-1, +1\}$ the class labels; $N = 4$ data points. (a) True posterior with $\log Z = -1.13$. (b) Optimal Gaussian approximation with bound value $\mathcal{B}_{G\text{-}KL} = -1.42$. (c) Optimal AI skew-normal approximation with bound value $\mathcal{B}_{AI\text{-}KL} = -1.17$.

D-dimensional integral as a one dimensional integral using the integral transform $\delta(x) = \int e^{2\pi i t x} dt$:

$$q_y(y) = \int\int e^{2\pi i t(y - \boldsymbol{\alpha}^\mathsf{T}\mathbf{v} - \beta)} \prod_{d=1}^{D} q_{v_d}(v_d) d\mathbf{v} dt = \int e^{2\pi i (t-\beta) y} \prod_{d=1}^{D} \tilde{q}_{u_d}(t)\, dt \tag{6.3.2}$$

where $\tilde{f}(t)$ denotes the Fourier transform of the function $f(x)$ and $q_{u_d}(u_d|\theta_d)$ is the density of the random variable $u_d := \alpha_d v_d$ so that $q_{u_d}(u_d|\theta_d) = \frac{1}{|\alpha_d|} q_{v_d}(\frac{u_d}{\alpha_d}|\theta_d)$. Equation (6.3.2) can be interpreted as the (shifted) inverse Fourier transform of the product of the Fourier transforms of $\{q_{u_d}(u_d|\theta_d)\}_{d=1}^D$.

Unfortunately, most distributions do not have Fourier transforms that admit compact analytic forms for the product $\prod_{d=1}^D \tilde{q}_{u_d}(t)$. The notable exception is the family of stable distributions for which linear combinations of random variables are also stable distributed – see Nolan [2012] for an introduction. With the exception of the Gaussian (the only stable distribution with finite variance), the Levy and the Cauchy distributions, stable distributions do not have analytic forms for their density functions and are analytically expressible only in the Fourier domain. Nevertheless, when $q_{\mathbf{v}}(\mathbf{v})$ is stable distributed, marginal quantities of $\mathbf{w}$ such as y can be computed analytically in the Fourier domain [Bickson and Guestrin, 2010].

In general, therefore, we need to resort to numerical methods to compute $q_y(y)$ and expectations with respect to it. To achieve this we discretise the base distributions and, by choosing a sufficiently fine discretisation, limit the maximal error that can be incurred. As such, up to a specified accuracy, the KL bound may be exactly computed.

First we define the set of discrete approximations to $\{q_{u_d}(u_d|\theta_d)\}_{d=1}^D$ for $u_d := \alpha_d v_d$. These 'lattice' approximations are a weighted sum of K delta functions

$$q_{u_d}(u_d|\theta_d) \approx \hat{q}_{u_d}(u_d) := \sum_{k=1}^{K} \pi_{dk} \delta(u_d - l_k) \quad \text{where} \quad \pi_{dk} = \int_{l_k - \frac{1}{2}\Delta}^{l_k + \frac{1}{2}\Delta} q(u_d|\theta_d) du_d. \tag{6.3.3}$$

The lattice points $\{l_k\}_{k=1}^K$ are spaced uniformly over the domain $[l_1, l_K]$ with $\Delta := l_{k+1} - l_k$. The

weighting for each delta spike is the mass assigned to the distribution $q_{u_d}(u_d|\theta_d)$ over the interval $[l_k - \frac{1}{2}\Delta, l_k + \frac{1}{2}\Delta]$.

Given the lattice approximations to the densities $\{q_{u_d}(u_d|\theta_d)\}_{d=1}^D$ the Fast Fourier Transform (FFT) can be used to evaluate the convolution of the lattice distributions. Doing so we obtain the lattice approximation to the marginal $y = \mathbf{w}^\mathsf{T}\mathbf{h}$ such that

$$q_y(y) \approx \hat{q}_y(y) = \sum_{k=1}^{K} \delta(y - l_k - \beta)\rho_k \quad \text{where} \quad \boldsymbol{\rho} = \texttt{ifft}\left[\prod_{d=1}^{D} \texttt{fft}\left[\boldsymbol{\pi}'_d\right]\right]. \tag{6.3.4}$$

where $\boldsymbol{\pi}_d$ is padded with $(D-1)K$ zeros, $\boldsymbol{\pi}'_d := [\boldsymbol{\pi}_d, \mathbf{0}]$. The only approximation used in finding the marginal density is then the discretisation of the base distributions, with the remaining FFT calculations being exact. See Appendix C.1.2 for a derivation of this result. The time complexity for the above procedure scales $O\left(D^2 K \log KD\right)$.

Efficient site derivative computation

Whilst we have shown that the expectation of the site-projections can be accurately computed using the FFT, how to cheaply evaluate the derivative of this term is less clear. The complication can be seen by inspecting the partial derivative of $\left\langle g(\mathbf{w}^\mathsf{T}\mathbf{h})\right\rangle$ with respect to A_{mn}

$$\frac{\partial}{\partial A_{mn}}\left\langle f(\mathbf{w}^\mathsf{T}\mathbf{h})\right\rangle_{q(\mathbf{w})} = h_n \int q_{\mathbf{v}}(\mathbf{v}) f'\left(\mathbf{h}^\mathsf{T}\mathbf{A}\mathbf{v} + \mathbf{b}^\mathsf{T}\mathbf{h}\right) v_m d\mathbf{v},$$

where $f'(y) = \frac{d}{dy}f(y)$. Naively, this can be readily reduced to a, relatively expensive, two dimensional integral. Critically, however, the computation can be simplified to a one dimensional integral. To see this we can write

$$\frac{\partial}{\partial A_{mn}}\left\langle f\left(\mathbf{w}^\mathsf{T}\mathbf{h}\right)\right\rangle = x_n \int f'(y) d_m(y) dy,$$

where

$$d_m(y) := \int v_m q_{\mathbf{v}}(\mathbf{v}) \delta\left(y - \boldsymbol{\alpha}^\mathsf{T}\mathbf{v} - \beta\right) d\mathbf{v}.$$

Here $d_m(y)$ is a univariate weighting function with Fourier transform:

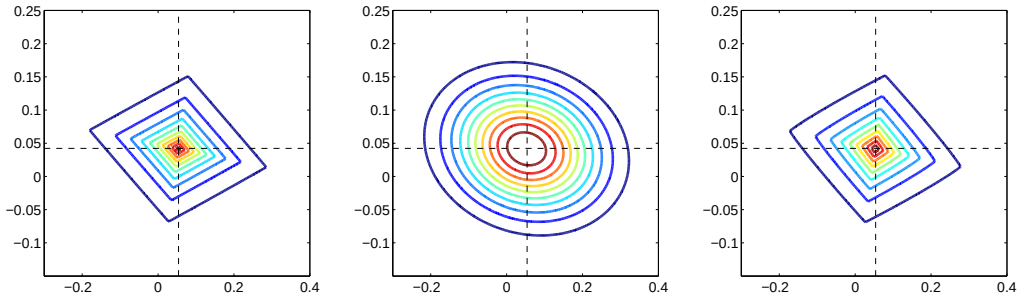
$$\tilde{d}_m(t) = e^{-2\pi i t\beta}\tilde{e}_m(t) \prod_{d\neq m} \tilde{q}_{u_d}(t), \quad \text{where} \quad \tilde{e}_m(t) := \int \frac{u_m}{\alpha_m} q_{u_m}(u_m) e^{-2\pi i t u_m} du_m.$$

Since $\{\tilde{q}(t)\}_{d=1}^D$ are required to compute the expectation of $\left\langle f(\mathbf{w}^\mathsf{T}\mathbf{h})\right\rangle$ the only additional computations needed to evaluate all partial derivatives with respect to $\mathbf{A}$ are $\{\tilde{e}_d(t)\}_{d=1}^D$. Thus the complexity of computing the site derivative is equivalent to the complexity of the site expectation of Section 6.3.1. Further derivations and computational scaling properties are provided in Appendix C.1.

6.3.2 Gaussian potential

For the Gaussian potential $\mathcal{N}(\mathbf{w}|\boldsymbol{\mu}, \boldsymbol{\Sigma})$, its log expectation under $q_{\mathbf{w}}(\mathbf{w})$ can be expressed as

$$2\left\langle \log \mathcal{N}(\mathbf{w}|\boldsymbol{\mu}, \boldsymbol{\Sigma})\right\rangle = -D\log 2\pi - \log\det(\boldsymbol{\Sigma}) - \left\langle \mathbf{w}^\mathsf{T}\boldsymbol{\Sigma}^{-1}\mathbf{w}\right\rangle + 2\left\langle \mathbf{w}\right\rangle \boldsymbol{\Sigma}^{-1}\boldsymbol{\mu} - \boldsymbol{\mu}^\mathsf{T}\boldsymbol{\Sigma}^{-1}\boldsymbol{\mu}, \tag{6.3.5}$$



(a) (b) (c)

Figure 6.3: Two dimensional robust linear regression with Gaussian prior $\mathcal{N}(\mathbf{w}|\mathbf{0}, \mathbf{I})$, Laplace likelihood $\phi_n(\mathbf{w}) = \frac{1}{2\tau_l} e^{-|y_n - \mathbf{w}^\mathsf{T}\mathbf{h}_n|/\tau_l}$ with $\tau_l = 0.1581$ and 2 data pairs $\mathbf{h}_n, y_n$. (a) True posterior with $\log Z = -3.5159$. (b) Optimal Gaussian approximation with bound value $\mathcal{B}_{G\text{-}KL} = -3.6102$. (c) Optimal AI generalised-normal approximation with bound value $\mathcal{B}_{AI\text{-}KL} = -3.5167$.

where analytic forms for the quadratic expectation $\langle \mathbf{w}^\mathsf{T}\boldsymbol{\Sigma}^{-1}\mathbf{w} \rangle$ and the linear expectation $\langle \mathbf{w} \rangle$ can be expressed in terms of the first and second order moments of the factorising base density $q_\mathbf{v}(\mathbf{v})$. Explicit forms for equation (6.3.5) are given in Appendix C.1.3. The moments of the skew-normal and generalised-normal base densities, for which closed form expression exist, used for the experiments in this chapter are presented in Appendix A.5. Thus the Gaussian potential's contribution to the A-KL bound, and its gradient, can be computed using standard matrix vector operations. For a Gaussian potential with unstructured covariance, computing its contribution to the AI-KL bound scales $O\left(D^3\right)$ simplifying to $O\left(D^2\right)$ for isotropic covariance such that $\boldsymbol{\Sigma} = \sigma^2 \mathbf{I}_D$.

6.4 Optimising the AI-KL bound

Given fixed base distributions, we can optimise the KL bound with respect to the parameters $\mathbf{A} = \mathbf{L}\mathbf{U}$ and $\mathbf{b}$. Provided $\{\phi_n\}_{n=1}^N$ are log-concave the KL bound is jointly concave with respect to $\mathbf{b}$ and either $\mathbf{L}$ or $\mathbf{U}$. This follows from an application of the concavity result we provided in the Gaussian KL bound – see Appendix C.2.

Using a similar approach to that presented in Section 6.3.1 we can also efficiently evaluate the gradients of the KL bound with respect to the parameters $\boldsymbol{\theta}$ that define the base distribution. These parameters $\boldsymbol{\theta}$ can control higher order moments of the approximating density $q(\mathbf{w})$ such as skewness and kurtosis. We can therefore jointly optimise over all parameters $\{\mathbf{A}, \mathbf{b}, \boldsymbol{\theta}\}$ simultaneously; this means that we can fully capitalise on the expressiveness of the AI distribution class, allowing us to capture non-Gaussian structure in $p(\mathbf{w})$.

In many modeling scenarios the best choice for $q_\mathbf{v}(\mathbf{v})$ will suggest itself naturally. For example, in Section 6.7.1 we choose the skew-normal distribution to approximate Bayesian logistic regression posteriors. For heavy-tailed posteriors that arise for example in robust or sparse Bayesian linear regression models, one choice is to use the generalised-normal as base density, which includes the Laplace and

Gaussian distributions as special cases. For other models, for instance mixed data factor analysis [Khan et al., 2010], different distributions for blocks of variables of $\{v_d\}_{d=1}^D$ may be optimal. However, in situations for which it is not clear how to select $q_{\mathbf{v}}(\mathbf{v})$, several different distributions can be assessed and then that which achieves the greatest lower-bound $\mathcal{B}_{KL}$ is preferred.

6.5 Numerical issues

The computational burden of the numerical marginalisation procedure described in Section 6.3.1 depends on the number of lattice points used to evaluate the convolved density function $q_y(y)$. For the results presented we implemented a simple strategy for choosing the lattice points $[l_1, ..., l_K]$. Lattice end points were chosen[2] such that $[l_1, l_K] = [-6\sigma_y, 6\sigma_y]$ where σ_y is the standard deviation of the random variable y: $\sigma_y^2 = \sum_d \alpha_d^2 \mathrm{var}(v_d)$. From Chebyshev's inequality, taking six standard deviation end points guarantees that we capture at least 97% of the mass of $q_y(y)$. In practice this proportion is often much higher since $q_y(y)$ is often close to Gaussian for $D \gg 1$. We fix the number of lattice points used during optimisation to suit our computational budget. To compute the final bound value we apply the simple strategy of doubling the number of lattice points until the evaluated bound changes by less than 10^{-3} [Bracewell, 1986].

Fully characterising the overall accuracy of the approximation as a function of the number of lattice points is complex, see Ruckdeschel and Kohl [2010], Schaller and Temnov [2008] for a related discussion. One determining factor is the condition number (ratio of largest and smallest eigenvalues) of the posterior covariance. When the condition number is large many lattice points are needed to accurately discretise the set of distributions $\{q_{u_d}(u_d|\theta_d)\}_{d=1}^D$ which increases the time and memory requirements.

One possible route to circumventing these issues is to use base densities that have analytic Fourier transforms (such as a mixture of Gaussians). In such cases the discrete Fourier transform of $q_y(y)$ can be directly evaluated by computing the product of the Fourier transforms of each $\{q_{u_d}(u_d|\theta_d)\}_{d=1}^D$. The implementation and analysis of this procedure is left for future work.

The computational bottleneck for AI inference, assuming $N > D$, arises from computing the expectation and partial derivatives of the N site-projections. For parameters $\mathbf{w} \in \mathbb{R}^D$ this scales $O\left(ND^2K \log DK\right)$. Whilst this might appear expensive it is worth considering it within the broader scope of lower-bound inference methods. As we showed in Chapter 4, exact Gaussian KL approximate inference has bound and gradient computations which scale $O\left(ND^2\right)$. Similarly, local variational bounding methods (see below) scale $O\left(ND^2\right)$ when implemented exactly.

6.6 Related methods

Another commonly applied technique to obtain a lower-bound for densities of the form of equation (6.1.1) is the local variational bounding procedure introduced in Section 3.8. Local variational bounding methods approximate the normalisation constant by bounding each non-conjugate term in the integrand, equation (6.1.2), with a form that renders the integral tractable. In Chapter 4 we showed that the Gaussian

[2] For symmetric densities $\{q_{u_d}(u_d|\theta_d)\}$ we arranged that their mode coincides with the central lattice point.

KL bound dominates the local bound in such models. Hence the AI-KL method also dominates the local and Gaussian KL methods.

Other approaches to increasing the flexibility of the approximating distribution class include expressing $q_{\mathbf{w}}(\mathbf{w})$ as a mixture distribution – see Section 3.10. However, computing the entropy of a mixture distribution is in general difficult. Whilst one may bound the entropy term [Gershman et al., 2012, Bishop et al., 1998], employing such additional bounds is undesirable since it limits the gains from using a mixture. Another recently proposed method to approximate integrals using mixtures is split variational inference which iterates between constructing soft partitions of the integral domain and bounding those partitioned integrals [Bouchard and Zoeter, 2009]. The partitioned integrals are approximated using local or Gaussian KL bounds. Our AI method is complementary to the split mean field method since one may use the AI-KL technique to bound each of the partitioned integrals and so achieve an improved bound. However, this procedure should only be considered if extremely high accuracy approximations are required since it is likely to be very computationally demanding.

6.7 Experiments

For the experiments below[3], AI-KL bound optimisation is performed using L-BFGS[4]. Gaussian KL inference is implemented in all experiments using the `vgai` package.

6.7.1 Toy problems

We compare Gaussian KL and AI-KL approximate inference methods in three, two-dimensional generalised linear models against the true posteriors and marginal likelihood values obtained numerically. Figure 6.1 presents results for a linear regression model with a sparse Laplace prior; the AI base density is chosen to be generalised-normal. Figure 6.2 demonstrates approximating a Bayesian logistic regression posterior, with the AI base distribution skew-normal. Figure 6.3 corresponds to a Bayesian linear regression model with the noise robust Laplace likelihood density and Gaussian prior; again the AI approximation uses the generalised-normal as the base distribution.

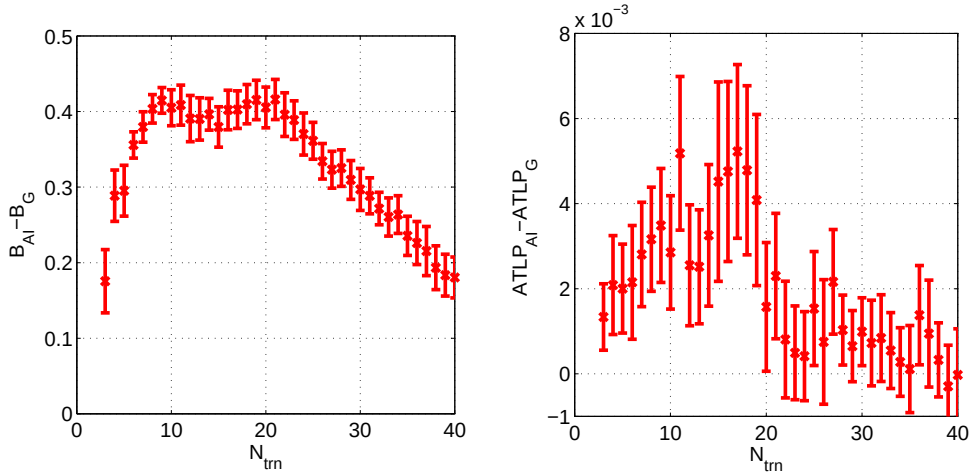
The AI-KL procedure achieves a consistently higher bound than the G-KL method, with the AI bound nearly saturating at the true value of $\log Z$ in two of the models. In addition, the AI approximation captures significant non-Gaussian features of the posterior: the approximate densities are sparse in directions of sparsity of the posterior; their modes are approximately equal (where the Gaussian mode can differ significantly); tail behaviour is more accurately captured by the AI distribution than by the Gaussian.

6.7.2 Bayesian logistic regression

We compare Gaussian KL and AI-KL approximate inference for a synthetic Bayesian logistic regression model. The AI density has skew-normal base distribution with θ_d parameterising the skewness of v_d. We optimised the AI-KL bound jointly with respect to $\mathbf{L}, \mathbf{U}, \mathbf{b}$ and $\boldsymbol{\theta}$ simultaneously with convergence

[3] All experiments are performed in Matlab 2009b on a 32 bit Intel Core 2 Quad 2.5 GHz processor.

[4] L-BFGS was implemented using the `minFunc` optimisation package (`www.di.ens.fr/~mschmidt`)



(a) (b)

Figure 6.4: Gaussian KL and AI-KL approximate inference comparison for a Bayesian logistic regression model with different training dataset sizes N_{trn}. $\mathbf{w} \in \mathbb{R}^{10}$; Gaussian prior $\mathcal{N}(\mathbf{w}|\mathbf{0}, 5\mathbf{I})$; logistic sigmoid likelihood $\phi_n = \sigma(\tau_l c_n \mathbf{w}^\mathsf{T}\mathbf{h}_n)$ with $\tau_l = 5$; covariates $\mathbf{h}_n$ sampled from the standard normal, $\mathbf{w}^{true}$ sampled from the prior and class labels $c_n = \pm 1$ sampled from the likelihood. (a) Bound differences, $\mathcal{B}_{AI\text{-}KL} - \mathcal{B}_{G\text{-}KL}$, achieved using Gaussian KL and AI-KL approximate inference for different training dataset sizes N_{trn}. Mean and standard errors are presented from 15 randomly generated models. A logarithmic difference of 0.4 corresponds to 49% improvement in the bound on the marginal likelihood. (b) Mean and standard error Averaged Testset Log Probability (ATLP) differences obtained with the Gaussian and AI approximate posteriors for different training dataset sizes N_{trn}. ATLP values calculated using 10^4 test data points sampled from each model.

taking on average 8 seconds with $D = N = 10$, compared to 0.2 seconds for Gaussian KL.[5]

In figure 6.4(a) we plot the performance of the KL bound for Gaussian $q_{\mathbf{w}}(\mathbf{w})$ versus the skew-normal AI $q_{\mathbf{w}}(\mathbf{w})$ as we vary the number of data points. We plot the mean and standard error bound differences $\mathcal{B}_{AI\text{-}KL} - \mathcal{B}_{G\text{-}KL}$ obtained over 15 randomly generated datasets. For a small number of data points the bound difference is small. This difference increases up to $D = N$, and then decreases for larger datasets. This behaviour can be explained by the fact that when there are few data points the Gaussian prior dominates, with little difference therefore between the Gaussian and optimal AI approximation (which becomes effectively Gaussian). As more data is introduced, the non-Gaussian likelihood terms have a stronger impact and the posterior becomes significantly non-Gaussian. However as even more data is introduced the central limit theorem effect takes hold and the posterior becomes increasingly Gaussian.

In figure 6.4(b) we plot the mean and standard error differences for the averaged testset log probabilities (ATLP) calculated using the Gaussian and AI approximate posteriors obtained in each model.

[5]We note that split mean field approximate inference was reported to take approximately 100 seconds for a similar logistic regression model achieving comparable results [Opper and Archambeau, 2009].

For each model and each training set size the ATLP is calculated using 10^4 test points sampled from the model. The log testset probability of each test data pair $\mathbf{h}^*, c^*$ is calculated as $\log \langle p(c^*|\mathbf{w}, \mathbf{h}^*) \rangle_{q(\mathbf{w})}$ for $q(\mathbf{w})$ the approximate posterior. The bound differences can be seen to be strongly correlated with testset log probability differences, seeming to confirm that tighter bound values correspond to improved predictive performance.

6.7.3 Sparse noise robust kernel regression

In this experiment we consider sparse noise robust kernel regression. Sparsity is encoded using a Laplace prior on the weight vectors $\prod_n p_n(w_n)$ where $p_n(w_n) = e^{-|w_n|/\tau_p}/2\tau_p$. The Laplace distribution is also used as a noise robust likelihood $\phi_n(\mathbf{w}) = p(y_n|\mathbf{w}, \mathbf{k}_n) = e^{-|y_n - \mathbf{w}^\mathsf{T}\mathbf{k}_n|/\tau_l}/2\tau_l$ where $\mathbf{k}_n$ is the n^{th} vector of the kernel matrix. For these experiments we use the isotropic squared exponential kernel such that

$$k_{m,n} := K(\mathbf{x}_m, \mathbf{x}_n, \kappa_l, \kappa_n) = e^{-\kappa_l \sum_{d=1}^{D}(x_{md} - x_{nd})^2} + \kappa_n \delta(n, m)$$

where $\delta(n, m)$ is the Kronecker delta function, x_{nd} is the d^{th} element of the n^{th} data point, the length scale parameter is set to $\kappa_l = 0.05$ and the additive noise is set to $\kappa_n = 1$. Thus the target density is defined as

$$p(\mathbf{w}|\mathcal{D}, \boldsymbol{\theta}) = \frac{1}{Z} \prod_{n=1}^{N_{trn}} \frac{1}{2\tau_p} e^{-\frac{|w_n|}{\tau_p}} \frac{1}{2\tau_l} e^{-\frac{\left|y_n - \mathbf{w}^\mathsf{T}\mathbf{k}_n\right|}{\tau_l}}, \tag{6.7.1}$$

where $\boldsymbol{\theta}$ denotes the collection of hyperparameters $\boldsymbol{\theta} := \{\tau_p, \tau_l, \kappa_l, \kappa_n\}$ and the vector of parameters is $\mathbf{w} \in \mathbb{R}^{N_{trn}}$. In all experiments the prior and likelihood were fixed with $\tau_p = \tau_l = 0.16$.

Three datasets were considered: Boston housing[6] ($D = 14$, $N_{trn} = 100$, $N_{tst} = 406$); Concrete Slump Test[7] ($D = 10$, $N_{trn} = 100$, $N_{tst} = 930$); and a synthetic dataset constructed as described in Kuss [2006] §5.6.1 ($D = 10$, $N_{trn} = 100$, $N_{tst} = 406$). Results are collected for each dataset over 10 random training and testset partitions. All datasets are zero mean unit variance normalised based on the statistics of the training data.

AI-KL inference is performed with a generalised-normal base distribution. The parameters θ_d control the kurtosis of the base distributions $q(v_d|\theta_d)$; for simplicity we fix $\theta_d = 1.5$ and optimise jointly for $\mathbf{L}, \mathbf{U}, \mathbf{b}$. Bound optimisation took roughly 250 seconds for the AI-KL procedure, compared to 5 seconds for the Gaussian KL procedure. Averaged results and standard errors are presented in Table 6.1 where $\bar{\mathcal{B}}_{KL}$ denotes the bound value divided by the number of points in the training dataset. Whilst the improvements for these particular datasets are modest, the AI bound dominates the Gaussian bound in all three datasets, with predictive log probabilities also showing consistent improvement.

Whilst we have only presented experimental results for AI distributions with simple analytically expressible base distributions we note the method is applicable for any base distribution provided $\{q_{v_d}(v_d)\}_{d=1}^{D}$ are smooth, a condition that is required to ensure differentiability of the AI-KL bound. For example smooth univariate mixtures for $q_{v_d}(v_d)$ can be used.

[6] archive.ics.uci.edu/ml/datasets/Housing

[7] archive.ics.uci.edu/ml/datasets/Concrete+Slump+Test

Dataset	$\bar{\mathcal{B}}_{G\text{-}KL}$	$\bar{\mathcal{B}}_{AI\text{-}KL}$	$\bar{\mathcal{B}}_{AI\text{-}KL} - \bar{\mathcal{B}}_{G\text{-}KL}$	$ATLP_G$	$ATLP_{AI}$	$ATLP_{AI} - ATLP_G$
Conc. CS.	-2.08 ± 0.09	-2.06 ± 0.09	0.022 ± 0.004	-1.70 ± 0.11	-1.67 ± 0.11	0.024 ± 0.010
Boston	-1.28 ± 0.05	-1.25 ± 0.05	0.028 ± 0.003	-1.18 ± 0.10	-1.15 ± 0.09	0.023 ± 0.006
Synthetic	-2.49 ± 0.10	-2.46 ± 0.10	0.028 ± 0.004	-1.84 ± 0.11	-1.83 ± 0.11	0.009 ± 0.009

Table 6.1: AI-KL approximate inference results for the sparse noise robust kernel regression model. The model is defined by a factorising Laplace prior on the weights such that $p(w_n) = e^{-|w_n|/\tau_p}/2\tau_p$ and a Laplace conditional likelihood $p(y_n|\mathbf{w}^\mathsf{T}\mathbf{k}_n) = e^{-|y_n - \mathbf{w}^\mathsf{T}\mathbf{k}_n|/\tau_l}/2\tau_l$, with $\tau_p = \tau_l = 0.16$ and a squared exponential kernel. Values are the mean and standard error scores obtained from 10 random training and testset splits. $\bar{\mathcal{B}}_{G\text{-}KL}$ and $\bar{\mathcal{B}}_{AI\text{-}KL}$ denote the log marginal likelihood KL bound values, normalised by dividing by the size of the dataset N_{trn}, achieved using the Gaussian or AI variational densities. Averaged Testset Log Probability ($ATLP$) scores are calculated using $ATLP = \frac{1}{N_{tst}} \sum_n \log \langle p(y_n^*|\mathbf{w}, \mathbf{k}_n^*) \rangle_{q(\mathbf{w})}$.

6.8 Summary

Affine independent KL approximate inference has several desirable properties compared to existing deterministic bounding methods. We have shown how it generalises on classical multivariate Gaussian KL approximations and our experiments confirm that the method is able to capture non-Gaussian effects in posteriors. Since we optimise the KL divergence over a larger class of approximating densities than the multivariate Gaussian, the lower-bound to the normalisation constant is also improved. This is particularly useful for model selection purposes where the normalisation constant plays the role of the model likelihood.

6.9 Future work

The AI-KL approximate inference procedure proposed here poses several interesting directions for further research. The numerical procedures presented in Section 6.3 provide a general and computationally efficient means for inference in non-Gaussian densities whose application could be useful for a range of probabilistic models. However, our current understanding of the best approach to discretise the base densities is limited and further study of this is required. Furthermore, optimisation was found to be slow compared to G-KL procedures. Whilst this is not surprising, the AI-KL seeks a more accurate approximation than a Gaussian and requires optimising more than twice as many parameters, it would remain highly beneficial to develop faster optimisation routines.

Numerical errors

The numerical procedure we introduced to evaluate the marginal density $y := \mathbf{w}^\mathsf{T}\mathbf{h}$ for $\mathbf{w} \sim q_\mathbf{w}(\mathbf{w})$ an AI distributed random variable introduces three separate sources for numerical error which we detail below. We take this analysis from Schaller and Temnov [2008].

truncation error The lattice approximations $\hat{p}(u_d)$ have bounded support. Provided we have analytic forms for the densities $p(v_d)$ the probability mass that is truncated can be assessed. For example limit points $[l_0, l_K]$ can chosen such that $1 - \int_{l_0}^{l_K} p(u_d)du_d < 10^{-6}$ for all d. For heavy tailed densities limit points that satisfy such a condition may be infeasibly far apart.

aliasing error Discrete convolution algorithms perform a cyclic convolution. However, since $\boldsymbol{\pi}_d$ is not periodic it must be padded with DK zeros to remove aliasing error completely. Full zero padding can be computationally expensive thus allowing for a small amount of aliasing error may be computationally necessary.

discretisation error Discretisation error is introduced at the numerical convolution step required to calculate the marginal density. The analysis of discretisation error is more involved than the aliasing and truncation errors.

The first proposed direction for future research is to better understand how these three sources of numerical error described above depend on the parameters of the AI density $\mathbf{A}, \mathbf{b}$, the base density class $\{q_{v_d}(v_d|\theta_d)\}$ and the marginal projection vector $\mathbf{h}$. One possible relation between these factors, that was observed in our experimental work, was discussed in Section 6.5. Analysis of these errors is complex and would presumably require some sophistication in numerical analysis.

The second direction for future work is to develop methods to reduce these errors. For example, for heavy tailed densities truncation error can be reduced by using 'exponential windowing' methods that pre-transform the marginals $\{q(u_d|\theta_d)\}$ by an exponentially decaying function and then invert the transform after convolution [Schaller and Temnov, 2008]. Reduced aliasing and discretisation error could possibly also be achieved by constraining the class of base densities.

Another possible direction of work to increase the accuracy of the marginal evaluation and reduce complexity of this computation is to consider discretisations performed in the Fourier domain. For example, it might be possible to construct a discrete approximation directly on $\tilde{q}(y)$, as defined in equation (6.3.2), and numerically invert that approximation. Such a procedure, if feasible, could possibly reduce the effects of discretisation and aliasing error whilst reducing the computational complexity.

Chapter 7

Summary and conclusions

Latent linear models are widely used and form the backbone of many machine learning and computational statistics methods. Latent linear models are employed principally for their simplicity and representational power. As discussed in Chapter 2, performing inference in this model class has numerous advantages over simple point estimation techniques. However, beyond the most simple, fully Gaussian latent linear models, exact analytic forms for the inferential quantities of interest can rarely be derived and so approximations are required. One approach to approximate inference is to use sampling based methods such as Monte Carlo Markov chain. Whilst sampling techniques are widely applicable and can be highly accurate, assessing convergence can be difficult. An alternative approach to approximate inference is to use deterministic variational methods. Deterministic methods can exploit the highly structured form of the latent linear model class to provide relatively accurate approximate inferences quickly. Often, speed and accuracy of inference are critical if we are to employ the latent linear model class in real world applications. It is to this end that we seek to develop fast, accurate and widely applicable deterministic approximate inference methods for latent linear models.

In Chapter 2 we briefly introduced, reviewed and motivated the need for and uses of the inferential quantities $p(\mathbf{w})$ and Z in latent linear models. In Chapter 3 we provided an introduction and overview of the most commonly used deterministic approximate inference methods in the latent linear model class. In chapters 4, 5 and 6 we presented our core contributions to this problem domain. Below we briefly review and summarise these contributions.

7.1 Gaussian Kullback-Leibler approximate inference

In Chapter 4 we considered a method to obtain a Gaussian approximation to a latent linear model target density $p(\mathbf{w})$, and a lower-bound on the target density's normalisation constant Z, by minimising the Kullback-Leibler divergence between the two distributions. We referred to this procedure as the Gaussian Kullback-Leibler (G-KL) approximation. As we saw in Chapter 3, G-KL methods have been known about for some time but have received comparatively little attention from the research community. Principally this was because of the perceived computational complexity of G-KL procedures.

Previous authors advocated optimising the G-KL bound using a particular constrained form of covariance which we described in Section 4.3.1.1. However, as discussed in Chapter 4, this parameterisa-

tion requires multiple cubic matrix operations to evaluate the G-KL bound and its derivative and renders the bound non-concave. Thus G-KL procedures using this parameterisation are both inefficient and unscalable. In Chapter 4 we showed that these problems can be addressed by parameterising the G-KL bound with respect to the Cholesky decomposition of covariance. For Cholesky parameterisations of G-KL covariance we showed that bound and gradient computations could be performed more efficiently and that the bound was concave for this parameterisation. Furthermore, we provided constrained parameterisations of covariance that made the G-KL method scalable. These developments make G-KL approximate inference methods one of the most efficient and scalable deterministic Gaussian approximate inference methods in the latent linear model class. As our numerical experiments in Chapter 5 show, G-KL procedures are highly competitive compared to other deterministic Gaussian approximations with regards to accuracy, speed and scalability.

As we saw in Chapter 3, many deterministic approximate inference methods provide a Gaussian approximation to the target density and at least two other methods, local variational bounding and mean field bounding, provide a lower bound on Z. Each of these methods differ, however, in the restrictions they place on the non-Gaussian potentials of the latent linear model, the speed and scalability properties of their optimisation procedures and the accuracy of their approximations. Local Variational Bounding (LVB) and G-KL methods provide a Gaussian approximation the target density, approximate the full covariance structure of the target and provide a lower-bound to the target's normalisation constant Z. Under which circumstances then, if any, should one of these methods be preferred over the other? Prior to our contributions, with regards to accuracy of inference, the answer to this question was unclear, and with regards to speed and scalability the answer was that LVB should be preferred. Previous empirical work suggested that the G-KL approach could be more accurate than LVB but was found to be much more computationally demanding. However, as we showed in Section 4.4.1, the G-KL procedure's lower-bound is guaranteed to dominate that provided by the LVB procedure. Furthermore, following our contributions, the results we provide in Chapter 5 show that G-KL procedures can be made as fast and as scalable as LVB methods in a range of problems, whilst also often dominating this method in terms of accuracy.

The primary strength of G-KL procedures is the ease with which they can be implemented and applied to new latent linear models. The only restriction G-KL methods place on the latent linear model, as described in Section 2.3, is that each of its non-Gaussian potentials $\{\phi_n\}$ has unbounded support. Unlike the Laplace approximation, G-KL does not require that the target is twice continuously differentiable. Unlike local variational bounding methods, G-KL does not require that each potential is super-Gaussian. Unlike expectation propagation methods, G-KL is numerically stable and does not require that each potential is log-concave. Unlike mean field methods, G-KL procedures can be applied to problems that do not satisfy the restrictive factorisation structure $\prod_n \phi_n(\mathbf{w}^\mathsf{T}\mathbf{h}_n) = \prod_d \phi_d(w_d)$. Indeed all that is required to apply G-KL methods to a linear model with a new potential function (that has unbounded support) is that we can numerically compute $\langle \log \phi(z) \rangle$ for z univariate normally distributed. For most potentials of practical interest this is equivalent to requiring that $\log \phi(z)$ can be efficiently evaluated pointwise. G-

KL methods do not require the derivation or construction of novel bounds, derivatives or expectations for each new potential function considered. Despite the simplicity and ease of implementation of the G-KL procedure, our results confirm that it is also one of the most accurate deterministic Gaussian approximate inference methods.

To aid future development and research into G-KL procedures and other areas we have developed and released the `vgai` Matlab package. The `vgai` package implements G-KL approximate inference for the latent linear model class using the methods proposed in this thesis. The `vgai` package is described in Appendix B.8 and can be downloaded from `mloss.org/software/view/308/`.

7.2 Affine independent Kullback-Leibler approximate inference

As we saw in Chapter 3, the majority of popular deterministic approximate inference methods construct a coarse, highly structured approximation to the target density, namely a delta function approximation, a fully factorised approximation or a multivariate Gaussian approximation. However, in some contexts it may be important to more accurately approximate the finer structure of the target density. In such a setting we may be willing to sacrifice speed in exchange for increased accuracy of inference. It was this motivation, combined with the successes that had been achieved with Gaussian Kullback-Leibler procedures, that lead us to develop techniques to expand the class of variational approximating densities beyond the multivariate Gaussian.

Previous approaches to increasing the accuracy of deterministic approximate inference methods focused on using mixture densities to approximate the target density – see the discussion presented in Section 3.10. Deterministic mixture model approximations can be obtained using either mixture mean field methods or split variational inference methods. Mixture mean field methods optimise the KL lower-bound on Z with respect to $q(\mathbf{w})$ a mixture. To do this an additional bound on the entropy of the mixture density is required to make the computations tractable. Whilst increasing the representational power of the approximating density, this approach further weakens the bound on Z and requires the computation of $O\left(K^2\right)$ expectations to evaluate the bound on the entropy and, so, can be quite slow, where K is the number of mixture components. Split variational inference is implemented using a double loop algorithm that requires K mean field or G-KL optimisations for the inner loop and optimises the soft partition of the integral domain in the outer loop. Due to this double loop structure, split variational inference methods may also be quite slow.

In Chapter 6 we proposed to optimise the KL divergence over a class of multivariate densities that are constructed as the affine transformation of a fully factorised density – the Affine Independent density class. To make the Affine Independent KL (AI-KL) approximation tractable we developed a novel efficient numerical procedure, using the Fast Fourier Transform, to evaluate and optimise the KL bound. The resulting AI-KL variational inference procedure can be interpreted as a means to learn the basis of a factorising mean field like approximation.

Since the Gaussian is a special case of the AI density class, the AI-KL method is guaranteed to provide approximate inferences at least as good as standard G-KL procedures, and thus also local variational bounding methods. The numerical results we provided showed that the AI-KL approach is able to

capture non-Gaussian properties of target densities and consequently make more accurate inferences and predictions. Additionally, if required, the AI-KL procedure can be used in conjunction with the mixture mean field and split variational procedures to improve the accuracy of those methods in latent linear model inference problems.

Appendix A

Useful results

In this appendix we present general results that are made use of throughout the thesis: in Appendix A.1 we present information theoretic results and definitions, in Appendix A.2 we present results for Gaussian distributed random variables, in Appendix A.3 we introduce the expectation maximisation algorithm and log likelihood gradient ascent techniques to perform parameter estimation in latent variable models, in Appendix A.4 we briefly introduce the exponential family, in Appendix A.5 we provide equations for each of the potential functions used in this thesis and implemented in the `vgai` G-KL approximate inference Matlab package, in Appendix A.6 we provide various identities from linear algebra.

A.1 Information theory

Below we define both the differential Shannon entropy and the differential Kullback-Leibler divergence. Additional results regarding these quantities can be found in Cover and Thomas [1991].

A.1.1 Entropy

Shannon's entropy is a measure of the uncertainty of a random variable. For a continuous random variable $\mathbf{w}$ with density function $p(\mathbf{w})$, Shannon's differential entropy is defined as

$$H[p(\mathbf{w})] := -\int_{\mathcal{W}} p(\mathbf{w}) \log p(\mathbf{w}) d\mathbf{w},$$

where $\mathcal{W}$ denotes the support of $p(\mathbf{w})$ such that $\mathcal{W} := \left\{\mathbf{w} \in \mathbb{R}^D | p(\mathbf{w}) > 0\right\}$. Unlike the discrete entropy of a discrete random variable, the differential entropy is unbounded and can be either positive or negative.

Linear transformation. If we linearly transform the continuous random variable $\mathbf{w}$ such that $\tilde{\mathbf{w}} = \mathbf{A}\mathbf{w} + \mathbf{b}$, with $\mathbf{A} \in \mathbb{R}^{D \times D}$ non-singular and $\mathbf{b} \in \mathbb{R}^D$, it is easy to show that the entropy of $\tilde{\mathbf{w}}$ can be expressed as

$$H[p(\tilde{\mathbf{w}})] = H[p(\mathbf{w})] + \log |\det(\mathbf{A})|.$$

A.1.2 Conditional entropy

The conditional entropy is a measure of the uncertainty of one random variable conditioned on another. For a joint density $p(\mathbf{w}, \mathbf{v})$, the conditional differential entropy of $\mathbf{w}$ given $\mathbf{v}$ is defined as

$$H[p(\mathbf{w}|\mathbf{v})] := -\int\int p(\mathbf{w}, \mathbf{v}) \log p(\mathbf{w}|\mathbf{v}) d\mathbf{v} d\mathbf{w} = \int p(\mathbf{v}) H[p(\mathbf{w}|\mathbf{v})] d\mathbf{v}.$$

The conditional entropy is upper bounded by the marginal entropy such that $H[p(\mathbf{w}|\mathbf{v})] \leq H[p(\mathbf{w})]$, with equality if and only if $\mathbf{w}$ and $\mathbf{v}$ are independent.

A.1.3 Kullback-Leibler divergence

The Kullback-Leibler (KL) divergence, otherwise termed the relative entropy, is a measure the statistical difference between to random variables. The KL divergence is discussed at greater depth in Section 3.2. For the probability density functions $p(\mathbf{w})$ and $q(\mathbf{w})$ the differential KL divergence is defined as

$$\mathrm{KL}(q(\mathbf{w})|p(\mathbf{w})) = \int_{\mathcal{W}} q(\mathbf{w}) \log \frac{q(\mathbf{w})}{p(\mathbf{w})} d\mathbf{w},$$

where $\mathcal{W}$ denotes the support of $q(\mathbf{w})$. Note that the KL divergence can only be finite provided the support of $q(\mathbf{w})$ is contained in the support of $p(\mathbf{w})$.

The KL divergence between two multivariate Gaussian density functions $q(\mathbf{w}) = \mathcal{N}\left(\mathbf{w}|\boldsymbol{\mu}_q, \boldsymbol{\Sigma}_q\right)$ and $p(\mathbf{w}) = \mathcal{N}\left(\mathbf{w}|\boldsymbol{\mu}_p, \boldsymbol{\Sigma}_p\right)$ is given by the equality

$$2\mathrm{KL}(q(\mathbf{w})|p(\mathbf{w})) = \mathrm{trace}\left(\boldsymbol{\Sigma}_p^{-1}\boldsymbol{\Sigma}_q\right) + (\boldsymbol{\mu}_p - \boldsymbol{\mu}_q)^{\mathsf{T}}\boldsymbol{\Sigma}_p^{-1}(\boldsymbol{\mu}_p - \boldsymbol{\mu}_q) - D + \log\det\left(\boldsymbol{\Sigma}_p\right) - \log\det\left(\boldsymbol{\Sigma}_q\right).$$

A.2 Gaussian random variables

Gaussian random variables have various unique analytic and computational properties versus other continuous unbounded random variables. Below we introduce the univariate and multivariate Gaussian forms and describe some of the results that we make use of throughout the thesis. We note that 'Gaussian' and 'normal' are synonymous for the random variable described here.

A.2.1 Univariate Gaussian

For $v \in \mathbb{R}$ a univariate Gaussian standard normal (zero mean and unit variance) random variable, $v \sim \mathcal{N}(0,1)$, has its density function given by

$$p(v) = \frac{1}{\sqrt{2\pi}} e^{-\frac{1}{2}v^2} =: \mathcal{N}(v|0,1).$$

The linear transformation of v such that $w = \sigma v + \mu$, with $\sigma \in \mathbb{R}^+$ and $\mu \in \mathbb{R}$ fixed, is also Gaussian distributed, the density of w is given by

$$p(w|\mu,\sigma) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(w-\mu)^2}{2\sigma^2}} =: \mathcal{N}\left(w|\mu,\sigma^2\right),$$

where μ is the mean of w and σ^2 is its variance.

The univariate Gaussian density $\mathcal{N}\left(w|\mu,\sigma^2\right)$ admits the gradients

$$\frac{\partial}{\partial\mu}\mathcal{N}\left(w|\mu,\sigma^2\right) = \frac{(w-\mu)}{\sigma^2}\mathcal{N}\left(w|\mu,\sigma^2\right), \quad \text{and} \quad \frac{\partial}{\partial\sigma^2}\mathcal{N}\left(w|\mu,\sigma^2\right) = \frac{(w-\mu)^2}{\sigma^4}\mathcal{N}\left(w|\mu,\sigma^2\right).$$

A.2.2 Multivariate Gaussian

The joint density of a collection of D independent univariate Gaussian standard normal random variables $\mathbf{v} := [v_1, ..., v_D]^{\mathsf{T}}$ is defined by the product of the D univariate densities such that

$$p(v_1, ..., v_D) = \prod_{d=1}^{D} \frac{1}{\sqrt{2\pi}} e^{-v_d^2} = \frac{1}{(2\pi)^{D/2}} e^{-\frac{1}{2}\mathbf{v}^{\mathsf{T}}\mathbf{v}} = p(\mathbf{v}) =: \mathcal{N}(\mathbf{v}|\mathbf{0}, \mathbf{I}_D).$$

If we linearly transform $\mathbf{v}$ such that $\mathbf{w} = \mathbf{C}^\mathsf{T}\mathbf{v} + \boldsymbol{\mu}$ where $\mathbf{C} \in \mathbb{R}^{D\times D}$ is non-singular and $\boldsymbol{\mu} \in \mathbb{R}^D$, the density of $\mathbf{w}$ is given by

$$p(\mathbf{w}|\boldsymbol{\mu}, \mathbf{C}) = \frac{1}{\sqrt{2\pi \det(\boldsymbol{\Sigma})}} e^{-\frac{1}{2}(\mathbf{w}-\boldsymbol{\mu})\boldsymbol{\Sigma}^{-1}(\mathbf{w}-\boldsymbol{\mu})} =: \mathcal{N}(\mathbf{w}|\boldsymbol{\mu}, \boldsymbol{\Sigma}),$$

where we let $\boldsymbol{\Sigma} = \mathbf{C}^\mathsf{T}\mathbf{C}$. Therefore $\boldsymbol{\Sigma}$ is positive definite, and so without loss of generality we can restrict $\mathbf{C}$ to be upper-triangular with positive diagonal elements. For symmetric positive definite $\boldsymbol{\Sigma}$ such that $\boldsymbol{\Sigma} = \mathbf{C}^\mathsf{T}\mathbf{C}$, with $\mathbf{C}$ upper-triangular, then $\mathbf{C}$ is the unique Cholesky factorisation of $\boldsymbol{\Sigma}$ – see Appendix A.6.1 for further details of this factorisation.

Whitening data. The converse of the result above provides a means to 'whiten' data. For multivariate Gaussian random variables $\mathbf{x} \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ the data can be 'whitened' by transforming it such that $\tilde{\mathbf{x}} = \mathbf{C}^{-\mathsf{T}}(\mathbf{x} - \boldsymbol{\mu})$. Having performed this transformation, $\tilde{\mathbf{x}} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}_D)$. Whitening, is often used as a data pre-processing step to remove correlation from a set of covariates $\mathcal{D} = \{\mathbf{x}_n\}_{n=1}^N$ with $\boldsymbol{\Sigma}$ and $\boldsymbol{\mu}$ approximated using their empirical estimates.

Products and quotients of Gaussian densities. The product of two multivariate Gaussian densities is an unnormalised Gaussian density. Defining the product

$$\mathcal{N}(\mathbf{w}|\boldsymbol{\mu}_1, \boldsymbol{\Sigma}_1)\mathcal{N}(\mathbf{w}|\boldsymbol{\mu}_2, \boldsymbol{\Sigma}_2) = Z\mathcal{N}(\mathbf{w}|\boldsymbol{\mu}, \boldsymbol{\Sigma}),$$

then we have:

$$\boldsymbol{\Sigma} = \left(\boldsymbol{\Sigma}_1^{-1} + \boldsymbol{\Sigma}_2^{-1}\right)^{-1}, \quad \boldsymbol{\mu} = \boldsymbol{\Sigma}\left(\boldsymbol{\Sigma}_1^{-1}\boldsymbol{\mu}_1 + \boldsymbol{\Sigma}_2^{-1}\boldsymbol{\mu}_2\right), \quad \text{and} \quad Z = \mathcal{N}(\boldsymbol{\mu}_1|\boldsymbol{\mu}_2, \boldsymbol{\Sigma}).$$

The equations above can be used to derive moments of the Gaussian defined by the quotient $\mathcal{N}(\mathbf{w}|\boldsymbol{\mu}, \boldsymbol{\Sigma}) \propto \mathcal{N}(\mathbf{w}|\boldsymbol{\mu}_1, \boldsymbol{\Sigma}_1)/\mathcal{N}(\mathbf{w}|\boldsymbol{\mu}_2, \boldsymbol{\Sigma}_2)$ that is required, for example, when calculating Gaussian cavity densities during Gaussian expectation propagation optimisation procedures – see Section 3.6.

Normalisation constant. An exponentiated quadratic potential defined as

$$\phi(\mathbf{w}) \propto e^{-\frac{1}{2}\mathbf{w}^\mathsf{T}\mathbf{A}\mathbf{w} + \mathbf{w}^\mathsf{T}\mathbf{b}}, \tag{A.2.1}$$

with $\mathbf{A} \in \mathbb{R}^{D\times D}$ symmetric positive definite and $\mathbf{b} \in \mathbb{R}^D$, defines an unnormalised multivariate Gaussian density. The normalisation constant of $p(\mathbf{w}) = \phi(\mathbf{w})/Z$ can be evaluated by completing the square in the exponent of equation (A.2.1) and equating terms with the multivariate Gaussian. Doing so we see that

$$p(\mathbf{w}) = \mathcal{N}\left(\mathbf{w}|\mathbf{A}^{-1}\mathbf{b}, \mathbf{A}^{-1}\right), \quad \text{and} \quad Z := \int e^{-\frac{1}{2}\mathbf{w}^\mathsf{T}\mathbf{A}\mathbf{w} + \mathbf{w}^\mathsf{T}\mathbf{b}} d\mathbf{w} = \det(2\pi\mathbf{A})^{\frac{1}{2}} e^{\frac{1}{2}\mathbf{b}^\mathsf{T}\mathbf{A}^{-1}\mathbf{b}}.$$

Independencies. Zeros in the covariance matrix of a multivariate Gaussian density encode marginal independence relations, such that w_i is marginally independent of w_j iff $\Sigma_{ij} = 0$. Zeros in the precision matrix of a multivariate Gaussian density, where the precision matrix is defined as $\boldsymbol{\Gamma} := \boldsymbol{\Sigma}^{-1}$, encode conditional independence relations, so that w_i is independent of w_j conditioned on the remaining variables $\mathbf{w}_{\setminus\{i,j\}}$ iff $\Gamma_{ij} = 0$.

A.2.3 Gaussian inference

For a multivariate Gaussian density $p(\mathbf{w}) := \mathcal{N}(\mathbf{w}|\boldsymbol{\mu}, \boldsymbol{\Sigma})$, if we re-order and partition the variables $\mathbf{w}$ such that $\mathbf{w}^\mathsf{T} = \left[\mathbf{w}_1^\mathsf{T}, \mathbf{w}_2^\mathsf{T}\right]$, and similarly shuffle the parameters $\boldsymbol{\mu}$ and $\boldsymbol{\Sigma}$ according to the same index partition so that $\boldsymbol{\mu}^\mathsf{T} = \left[\boldsymbol{\mu}_1^\mathsf{T}, \boldsymbol{\mu}_2^\mathsf{T}\right]$ and

$$\boldsymbol{\Sigma} = \begin{bmatrix} \boldsymbol{\Sigma}_{11} & \boldsymbol{\Sigma}_{12} \\ \boldsymbol{\Sigma}_{21} & \boldsymbol{\Sigma}_{22} \end{bmatrix},$$

then marginals such as $p(\mathbf{w}_1)$ and conditionals such as $p(\mathbf{w}_1|\mathbf{w}_2)$ are Gaussian densities also. Below we specify the mean and covariance for these Gaussian marginal and conditional densities.

Gaussian marginal. The density of the marginal $p(\mathbf{w}_1)$ is multivariate Gaussian with

$$p(\mathbf{w}_1) = \int p(\mathbf{w}) d\mathbf{w}_2 = \mathcal{N}(\mathbf{w}_1|\boldsymbol{\mu}_1, \boldsymbol{\Sigma}_{11}).$$

Gaussian conditional. The density of $\mathbf{w}_1$ conditioned on $\mathbf{w}_2$ is multivariate Gaussian with

$$p(\mathbf{w}_1|\mathbf{w}_2) = \mathcal{N}\left(\mathbf{w}_1|\boldsymbol{\mu}_1 + \boldsymbol{\Sigma}_{12}\boldsymbol{\Sigma}_{22}^{-1}(\mathbf{w}_2 - \boldsymbol{\mu}_2), \boldsymbol{\Sigma}_{11} - \boldsymbol{\Sigma}_{12}\boldsymbol{\Sigma}_{22}^{-1}\boldsymbol{\Sigma}_{21}\right).$$

A.2.4 Gaussian expectations

For multivariate Gaussian random variables, $\mathbf{w} \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$, the expectations of many functions $f(\mathbf{w})$, $\langle f(\mathbf{w}) \rangle_{\mathcal{N}(\mathbf{w}|\boldsymbol{\mu},\boldsymbol{\Sigma})}$, admit compact analytic forms. Below we give a few examples of such functions that we make use of in the thesis.

Quadratic

For $f(\mathbf{w})$ a quadratic in $\mathbf{w}$, such that $f(\mathbf{w}) := \mathbf{w}^\mathsf{T}\mathbf{A}\mathbf{w} + \mathbf{b}^\mathsf{T}\mathbf{w}$, its Gaussian expectation with respect to $\mathcal{N}(\mathbf{w}|\boldsymbol{\mu}, \boldsymbol{\Sigma})$ given by

$$\left\langle \mathbf{w}^\mathsf{T}\mathbf{A}\mathbf{w} + \mathbf{b}^\mathsf{T}\mathbf{w} \right\rangle_{\mathcal{N}(\mathbf{w}|\boldsymbol{\mu},\boldsymbol{\Sigma})} = \boldsymbol{\mu}^\mathsf{T}\mathbf{A}\boldsymbol{\mu} + \mathrm{trace}(\mathbf{A}\boldsymbol{\Sigma}) + \mathbf{b}^\mathsf{T}\boldsymbol{\mu}.$$

Non-Gaussian univariate potential

For $f(\mathbf{w})$ a site projection potential such that $f(\mathbf{w}) = g(\mathbf{w}^\mathsf{T}\mathbf{h})$ where $g : \mathbb{R} \to \mathbb{R}$ and $\mathbf{h} \in \mathbb{R}^D$ a fixed vector, the expectation $\langle g(\mathbf{w}^\mathsf{T}\mathbf{h}) \rangle_{\mathcal{N}(\mathbf{w}|\boldsymbol{\mu},\boldsymbol{\Sigma})}$ can be expressed as a univariate expectation such that $\langle g(\mathbf{w}^\mathsf{T}\mathbf{h}) \rangle_{\mathcal{N}(\mathbf{w}|\boldsymbol{\mu},\boldsymbol{\Sigma})} = \langle g(y) \rangle_{\mathcal{N}(y|\boldsymbol{\mu}^\mathsf{T}\mathbf{h}, \mathbf{h}^\mathsf{T}\boldsymbol{\Sigma}\mathbf{h})}$. This result is relied upon to derive efficient Gaussian KL bound evaluation and optimisation routines in Chapter 4. The result is originally due to Barber and Bishop [1998b], we present it here for clarity of exposition.

We start by showing that the D-dimensional expectation $\langle g(\mathbf{w}^\mathsf{T}\mathbf{h}) \rangle_{\mathcal{N}(\mathbf{w}|\boldsymbol{\mu},\boldsymbol{\Sigma})}$ can be expressed as a univariate integral by making the substitution $g(\mathbf{w}^\mathsf{T}\mathbf{h}) = \int \delta(y - \mathbf{w}^\mathsf{T}\mathbf{h}) g(y) dy$

$$\begin{aligned}
\left\langle g(\mathbf{w}^\mathsf{T}\mathbf{h}) \right\rangle_{\mathcal{N}(\mathbf{w}|\boldsymbol{\mu},\boldsymbol{\Sigma})} &= \int \mathcal{N}(\mathbf{w}|\boldsymbol{\mu}, \boldsymbol{\Sigma})\, g(\mathbf{w}^\mathsf{T}\mathbf{h}) d\mathbf{w} \\
&= \int \mathcal{N}(\mathbf{w}|\boldsymbol{\mu}, \boldsymbol{\Sigma}) \int \delta(y - \mathbf{w}^\mathsf{T}\mathbf{h}) g(y) dy d\mathbf{w} \\
&= \int \underbrace{\int \mathcal{N}(\mathbf{w}|\boldsymbol{\mu}, \boldsymbol{\Sigma})\, \delta(y - \mathbf{w}^\mathsf{T}\mathbf{h}) d\mathbf{w}}_{:= p(y)} g(y) dy.
\end{aligned}$$

We now seek to show that $p(y) = \mathcal{N}\left(y|\boldsymbol{\mu}^\mathsf{T}\mathbf{h}, \mathbf{h}^\mathsf{T}\boldsymbol{\Sigma}\mathbf{h}\right)$. First we make the substitution $\mathbf{w} = \mathbf{C}^\mathsf{T}\mathbf{v} + \boldsymbol{\mu}$, where $\mathbf{C}$ is the Cholesky decomposition of $\boldsymbol{\Sigma}$ such that $\boldsymbol{\Sigma} = \mathbf{C}^\mathsf{T}\mathbf{C}$, to get

$$p(y) := \int \mathcal{N}\left(\mathbf{w}|\boldsymbol{\mu}, \boldsymbol{\Sigma}\right) \delta(y - \mathbf{w}^\mathsf{T}\mathbf{h}) d\mathbf{w} = \int \mathcal{N}\left(\mathbf{v}|\mathbf{0}, \mathbf{I}\right) \delta(y - \mathbf{v}^\mathsf{T}\mathbf{C}\mathbf{h} - \boldsymbol{\mu}^\mathsf{T}\mathbf{h}) d\mathbf{v}.$$

We now define a basis in the vector space $\mathbf{v}$ with unit normal basis vectors $\{\mathbf{e}_d\}_{d=1}^D$ such that $\mathbf{e}_1$ is parallel to $\mathbf{C}\mathbf{h}$ so $\mathbf{e}_1^\mathsf{T}\mathbf{C}\mathbf{h} = \|\mathbf{C}\mathbf{h}\|_2$ and $\mathbf{e}_d^\mathsf{T}\mathbf{C}\mathbf{h} = 0$ when $d \neq 1$. Since $\mathcal{N}\left(\mathbf{v}|\mathbf{0}, \mathbf{I}\right)$ is isotropic the density is invariant to orthonormal transformations $\mathcal{N}\left(\mathbf{v}|\mathbf{0}, \mathbf{I}\right) = \prod_{d=1}^D \mathcal{N}\left(\mathbf{e}_d^\mathsf{T}\mathbf{v}|0, 1\right)$ and so

$$\begin{aligned} p(y) &= \int \prod_{d=1}^D \mathcal{N}\left(v_d|0, 1\right) \delta(y - \sum_{d=1}^D v_d \mathbf{e}_d^\mathsf{T}\mathbf{C}\mathbf{h} - \boldsymbol{\mu}^\mathsf{T}\mathbf{h}) d\mathbf{v} \\ &= \int \mathcal{N}\left(v_1|0, 1\right) \delta(y - v_1\mathbf{e}_1^\mathsf{T}\mathbf{C}\mathbf{h} - \boldsymbol{\mu}^\mathsf{T}\mathbf{h}) dv_1 \\ &= \mathcal{N}\left(y|\boldsymbol{\mu}^\mathsf{T}\mathbf{h}, \|\mathbf{C}\mathbf{h}\|_2^2\right) = \mathcal{N}\left(y|\boldsymbol{\mu}^\mathsf{T}\mathbf{h}, \mathbf{h}^\mathsf{T}\boldsymbol{\Sigma}\mathbf{h}\right). \end{aligned}$$

Entropy

Shannon's differential entropy for multivariate Gaussian random variables, $\mathbf{w} \sim \mathcal{N}\left(\boldsymbol{\mu}, \boldsymbol{\Sigma}\right)$, is defined as the expectation of $-\log \mathcal{N}\left(\mathbf{w}|\boldsymbol{\mu}, \boldsymbol{\Sigma}\right)$ and has the analytic form

$$H\left[\mathcal{N}\left(\mathbf{w}|\boldsymbol{\mu}, \boldsymbol{\Sigma}\right)\right] := -\int \mathcal{N}\left(\mathbf{w}|\boldsymbol{\mu}, \boldsymbol{\Sigma}\right) \log \mathcal{N}\left(\mathbf{w}|\boldsymbol{\mu}, \boldsymbol{\Sigma}\right) d\mathbf{w} = \frac{1}{2} \log \det\left(2\pi e \boldsymbol{\Sigma}\right).$$

A.2.5 Gaussian density filter

Gaussian density filtering provides a means to approximate a density $p(\mathbf{w}) \propto \mathcal{N}\left(\mathbf{w}|\boldsymbol{\mu}, \boldsymbol{\Sigma}\right) \prod_n \phi(\mathbf{w}^\mathsf{T}\mathbf{h}_n)$, where ϕ_n are non-Gaussian potential functions, by a multivariate Gaussian $\mathcal{N}\left(\mathbf{w}|\mathbf{m}, \mathbf{S}\right)$. The approximation is obtained by iteratively re-approximating the 'tilted' density defined as $\tilde{p}(\mathbf{w}) = \mathcal{N}\left(\mathbf{w}|\mathbf{m}, \mathbf{S}\right) \phi(\mathbf{w}^\mathsf{T}\mathbf{h}_n)$ until all the non-Gaussian sites $\left\{\phi_n(\mathbf{w}^\mathsf{T}\mathbf{h}_n)\right\}_{n=1}^N$ have been 'included' in $\mathcal{N}\left(\mathbf{w}|\mathbf{m}, \mathbf{S}\right)$. The Gaussian $\mathcal{N}\left(\mathbf{w}|\mathbf{m}, \mathbf{S}\right)$ is initialised with $\mathbf{m} = \boldsymbol{\mu}$ and $\mathbf{S} = \boldsymbol{\Sigma}$. The results provided below follow the presentations made in [Minka, 2004, Herbrich, 2005].

It can be shown, see for example Minka [2001a], that the first two moments of $\tilde{p}(\mathbf{w})$ are given by

$$\begin{aligned} \tilde{\mathbf{m}} &:= \langle \mathbf{w} \rangle_{\tilde{p}(\mathbf{w})} = \mathbf{m} + \mathbf{S}\mathbf{g}, \\ \tilde{\mathbf{S}} &:= \left\langle (\mathbf{w} - \mathbf{m})(\mathbf{w} - \mathbf{m})^\mathsf{T} \right\rangle_{\tilde{p}(\mathbf{w})} = \mathbf{S} - \mathbf{S}\left(\mathbf{m}\mathbf{m}^\mathsf{T} - 2\mathbf{G}\right)\mathbf{S}, \end{aligned} \tag{A.2.2}$$

where the vector $\mathbf{g}$ and the matrix $\mathbf{G}$ are defined by

$$\mathbf{g} := \frac{\partial}{\partial \mathbf{m}} \log Z(\mathbf{h}, \mathbf{m}, \mathbf{S}) = \mathbf{h}\alpha(\mathbf{h}, \mathbf{m}, \mathbf{S}), \quad \text{and} \quad \mathbf{G} := \frac{\partial}{\partial \mathbf{S}} \log Z(\mathbf{h}, \mathbf{m}, \mathbf{S}) = \mathbf{h}\mathbf{h}^\mathsf{T}\gamma(\mathbf{h}, \mathbf{m}, \mathbf{S}),$$

where

$$\begin{aligned} Z(\mathbf{h}, \mathbf{m}, \mathbf{S}) &:= \int \phi(y) \mathcal{N}\left(y|m, s^2\right) dy, \\ \alpha(\mathbf{h}, \mathbf{m}, \mathbf{S}) &:= \frac{1}{Z(\mathbf{h}, \mathbf{m}, \mathbf{S})} \int \left(\frac{y - m}{s^2}\right) \phi(y) \mathcal{N}\left(y|m, s^2\right) dy, \\ \gamma(\mathbf{h}, \mathbf{m}, \mathbf{S}) &:= \frac{1}{Z(\mathbf{h}, \mathbf{m}, \mathbf{S})} \int \left(\frac{y - m}{s^2}\right)^2 \phi(y) \mathcal{N}\left(y|m, s^2\right) dy, \end{aligned} \tag{A.2.3}$$

and $m := \mathbf{m}^\mathsf{T}\mathbf{h}$, $s^2 := \mathbf{h}^\mathsf{T}\mathbf{S}\mathbf{h}$.

Note that in many cases the univariate expectations defined in equation (A.2.3) will have compact analytic forms. This is the case, for example, if $\phi(x)$ is a symmetric mixture of Heaviside step function or if $\phi(x)$ is the standard normal cumulative distribution function. When these expectations can not be evaluated analytically they can typically be computed efficiently using numerical integration routines [Zoeter and Heskes, 2005].

Using the mean and covariance as defined in equation (A.2.2), Gaussian density filtering recursively updates the approximating Gaussian moments by setting $\mathbf{m} \leftarrow \tilde{\mathbf{m}}$ and $\mathbf{S} \leftarrow \tilde{\mathbf{S}}$ until each non-Gaussian potential $\left\{\phi_n(\mathbf{w}^\mathsf{T}\mathbf{h}_n)\right\}$ has been 'included' in the approximation $\mathcal{N}(\mathbf{w}|\mathbf{m}, \mathbf{S})$. The Gaussian density filter is thus equivalent to a one pass sweep of the Gaussian EP updates – see Section 3.6.

A.3 Parameter estimation in latent variable models

In this section we present two approaches that are commonly used to perform parameter estimation in latent variable models: the Expectation Maximisation (EM) algorithm and log-likelihood gradient ascent. The EM and the gradient ascent procedures are closely related parameter estimation techniques with various authors having advocated methods that blend or switch between the two techniques to achieve more rapid convergence – for example see Jamshidian and Jennrich [1993], Salakhutdinov et al. [2003].

We consider the general form for the likelihood of a latent variable model to be defined as

$$p(\mathbf{v}|\boldsymbol{\theta}) = \int p(\mathbf{v}, \mathbf{h}|\boldsymbol{\theta}) d\mathbf{h},$$

where $\mathbf{v} \in \mathbb{R}^D$ denotes the visible variables, $\mathbf{h} \in \mathbb{R}^N$ denotes the hidden or latent variables and $\boldsymbol{\theta}$ is the set of parameters that we wish to optimise. Here we consider the special case where both the visible and hidden variables are continuous, for discrete hidden variables the integrals over $\mathbf{h}$ should be replaced by sums. The joint density $p(\mathbf{v}, \mathbf{h}|\boldsymbol{\theta})$ is often referred to as the complete likelihood. Here we make the additional assumption that it can be naturally decomposed into the factorisation $p(\mathbf{v}, \mathbf{h}|\boldsymbol{\theta}) = p(\mathbf{v}|\mathbf{h}, \boldsymbol{\theta}_v)p(\mathbf{h}|\boldsymbol{\theta}_h)$, where $\boldsymbol{\theta}^\mathsf{T} = [\boldsymbol{\theta}_v^\mathsf{T}, \boldsymbol{\theta}_h^\mathsf{T}]$.

For a dataset consisting of M data points, $\mathcal{D} = \{\mathbf{v}_m\}_{m=1}^M$, which are assumed independent and identically distributed (i.i.d.) given the parameters of the model $\boldsymbol{\theta}$, the likelihood of the data is given by the product

$$p(\mathcal{D}|\boldsymbol{\theta}) = \prod_{m=1}^M p(\mathbf{v}_m|\boldsymbol{\theta}) = \prod_{m=1}^M \int p(\mathbf{v}_m|\mathbf{h}_m, \boldsymbol{\theta}_v)p(\mathbf{h}_m|\boldsymbol{\theta}_h)d\mathbf{h}_m,$$

and the log likelihood of the data by the sum

$$\log p(\mathcal{D}|\boldsymbol{\theta}) = \sum_{m=1}^M \log p(\mathbf{v}_m|\boldsymbol{\theta}) = \sum_{m=1}^M \log \int p(\mathbf{v}_m|\mathbf{h}_m, \boldsymbol{\theta}_v)p(\mathbf{h}_m|\boldsymbol{\theta}_h)d\mathbf{h}_m. \tag{A.3.1}$$

Due to the log integral structure of the data log-likelihood, it is typically not easy to directly optimise equation (A.3.1) with respect to the parameters $\boldsymbol{\theta}$. Principally this is because the derivatives of equation (A.3.1) with respect to $\boldsymbol{\theta}$ will not, typically, admit simple analytic forms. Below we present two techniques that are often used to derive local optimisation procedures in latent variable models.

Expectation maximisation

The Expectation Maximisation (EM) algorithm is a general technique that can be used to perform parameter optimisation in latent variable models [Dempster et al., 1977]. The EM algorithm can be interpreted as a Kullback-Leibler lower-bound optimisation technique. To see this, first we consider the KL divergence between $p(\mathbf{h}|\mathbf{v}, \boldsymbol{\theta})$ and a variational density $q(\mathbf{h})$. Since $\mathrm{KL}(q(\mathbf{h})|p(\mathbf{h}|\mathbf{v}, \boldsymbol{\theta}))$ is positive for all densities $q(\mathbf{h})$ it provides the lower-bound

$$\log p(\mathbf{v}|\boldsymbol{\theta}) \geq H[q(\mathbf{h})] + \langle \log p(\mathbf{v}, \mathbf{h}|\boldsymbol{\theta}) \rangle_{q(\mathbf{h})} = H[q(\mathbf{h})] + \langle \log p(\mathbf{v}|\mathbf{h}, \boldsymbol{\theta}_v) \rangle_{q(\mathbf{h})} + \langle \log p(\mathbf{h}|\boldsymbol{\theta}_h) \rangle_{q(\mathbf{h})} ,$$

where $H[q(\mathbf{h})]$ is the differential entropy of $q(\mathbf{h})$. Since the log-likelihood of each data point can be bounded in this fashion, the log-likelihood of the data, equation (A.3.1), can be bounded using

$$\log p(\mathcal{D}|\boldsymbol{\theta}) \geq \underbrace{\sum_{m=1}^{M} H[q(\mathbf{h}_m)]}_{\text{entropy}} + \underbrace{\langle \log p(\mathbf{v}_m|\mathbf{h}_m, \boldsymbol{\theta}_v) \rangle_{q(\mathbf{h}_m)} + \langle \log p(\mathbf{h}_m|\boldsymbol{\theta}_h) \rangle_{q(\mathbf{h}_m)}}_{\text{energy}} . \tag{A.3.2}$$

The EM algorithm is an iterative, two stage, procedure to find a local optima of the log-likelihood. The EM algorithm iterates between performing the E-step and the M-step optimisations described below until convergence is achieved.

E-step During the E-step of the EM algorithm, equation (A.3.2) is optimised with respect to each of the variational densities $\{q(\mathbf{h}_m)\}_{m=1}^{M}$ with the parameters $\boldsymbol{\theta}$ held fixed. Differentiating equation (A.3.2) with respect to $q(\mathbf{h}_m)$, equating the derivative to zero, and on imposing normalisation constraints on $q(\mathbf{h}_m)$, we can see that equation (A.3.2) will be optimised when $q(\mathbf{h}_m) = p(\mathbf{h}_m|\mathbf{v}_m, \boldsymbol{\theta})$ for all $m = 1, \ldots, M$. Updating each of the variational densities in this manner is the exact E-step of the EM algorithm. Having performed an exact E-step, the bound in equation (A.3.2) saturates and is equal to the log-likelihood of the data at that parameter setting $\boldsymbol{\theta}$.

M-step During the M-step of the EM algorithm, equation (A.3.2) is optimised with respect to the parameters $\boldsymbol{\theta}$ with the variational densities $\{q(\mathbf{h}_m)\}_{m=1}^{M}$ held fixed. Ignoring constants with respect to the parameters $\boldsymbol{\theta}$ in equation (A.3.2), defines the 'energy' contribution to the bound, $E(\boldsymbol{\theta})$ such that

$$E(\boldsymbol{\theta}) = \sum_{m=1}^{M} \langle \log p(\mathbf{v}_m, \mathbf{h}_m|\boldsymbol{\theta}) \rangle_{q(\mathbf{h}_m)} = \sum_{m=1}^{M} \langle \log p(\mathbf{v}_m|\mathbf{h}_m, \boldsymbol{\theta}_v) \rangle_{q(\mathbf{h}_m)} + \langle \log p(\mathbf{h}_m|\boldsymbol{\theta}_h) \rangle_{q(\mathbf{h}_m)} .$$

An exact M-step corresponds to updating $\boldsymbol{\theta} := \boldsymbol{\theta}^*$ where $\boldsymbol{\theta}^* = \mathrm{argmax}_{\boldsymbol{\theta}} E(\boldsymbol{\theta})$. Often closed forms expressions can be derived for the M-step update. However, if this is not the case numerical procedures can also be used. Performing a partial M-step, where the parameters are updated to a setting that increases but does not optimise the energy, is referred to as the generalised EM algorithm. Both the generalised EM and the exact EM algorithm are guaranteed to increase the log-likelihood after each EM iteration.

Gradient ascent

The following well-known identity can be used to derive simple forms for the derivative of the log-likelihood in latent variable models

$$\begin{aligned}\frac{\partial}{\partial \boldsymbol{\theta}} \log p(\mathbf{v}|\boldsymbol{\theta}) &= \frac{1}{p(\mathbf{v}|\boldsymbol{\theta})} \frac{\partial}{\partial \boldsymbol{\theta}} p(\mathbf{v}|\boldsymbol{\theta}) = \frac{1}{p(\mathbf{v}|\boldsymbol{\theta})} \int \frac{\partial}{\partial \boldsymbol{\theta}} p(\mathbf{v}, \mathbf{h}|\boldsymbol{\theta}) d\mathbf{h} \\ &= \int p(\mathbf{h}|\mathbf{v}, \boldsymbol{\theta}) \frac{\partial}{\partial \boldsymbol{\theta}} \log p(\mathbf{v}, \mathbf{h}|\boldsymbol{\theta}) d\mathbf{h}, \end{aligned} \tag{A.3.3}$$

where we have made use of the relation that $(\log f(x))' = f'(x)/f(x)$. Thus evaluating the derivative, as expressed on the right hand side of the equation above, requires the evaluation of the latent variable conditional $p(\mathbf{h}|\mathbf{v}, \boldsymbol{\theta})$.

This identity can then be used to calculate the derivative of the log-likelihood of the data

$$\frac{\partial}{\partial \boldsymbol{\theta}} \log p(\mathcal{D}|\boldsymbol{\theta}) = \sum_{m=1}^{M} \left\langle \frac{\partial}{\partial \boldsymbol{\theta}} \log p(\mathbf{v}_m, \mathbf{h}_m|\boldsymbol{\theta}) \right\rangle_{p(\mathbf{h}_m|\mathbf{v}_m, \boldsymbol{\theta})}.$$

The derivative above can then be used to perform parameter estimation by gradient ascent of the log-likelihood. Similarly to the E-step of the EM algorithm, at each iteration of the gradient ascent procedure we need to infer the set of conditional densities $\{p(\mathbf{h}_m|\mathbf{v}_m, \boldsymbol{\theta})\}_{m=1}^{M}$.

Approximate EM algorithm

The EM algorithm can be relaxed by performing inexact E-steps. We refer to the EM algorithm with a partial or approximated E-step as the approximate EM algorithm. If a partial E-step is performed by optimising (but not maximising) equation (A.3.2) with respect to the variational densities $\{q(\mathbf{h}_m)\}_{m=1}^{M}$ the approximate EM algorithm is guaranteed to increase the lower-bound on the log-likelihood but not the log-likelihood itself. If the partial E-step is performed by updating the variational densities using some other approximation, for example the Laplace approximation, the approximate EM algorithm is not guaranteed to increase the log-likelihood or a lower bound on it.

A.4 Exponential family

A continuous random variable $\mathbf{w}$ is said to belong to the exponential family set of distributions if its density function can be expressed as

$$p(\mathbf{w}|\boldsymbol{\eta}) = g(\boldsymbol{\eta}) h(\mathbf{w}) \exp\left(\boldsymbol{\eta}^{\mathsf{T}} \mathbf{u}(\mathbf{w})\right), \tag{A.4.1}$$

where $\boldsymbol{\eta}$ is a vector collecting all the parameters of the density and is referred to as the distributions natural parameters. The function $\mathbf{u}(\mathbf{w})$ is a vector collecting the 'sufficient statistics' of the distribution and the function $g(\boldsymbol{\eta})$ ensures normalisation. Since the exponential family is linear with respect to the parameters $\boldsymbol{\eta}$ and the sufficient statistics $\mathbf{u}(\mathbf{w})$ in the exponent of equation (A.4.1), many simple analytic results can be derived for probabilistic models defined with respect to exponential family distributions. For example, below we show that $\frac{\partial}{\partial \boldsymbol{\eta}} - \log g(\boldsymbol{\eta}) = \langle \mathbf{u}(\mathbf{w}) \rangle_{p(\mathbf{w}|\boldsymbol{\eta})}$ which can be used to derive maximum likelihood estimates to $\boldsymbol{\eta}$ and fixed point updates for expectation propagation approximate inference routines – see Section 3.6.

To see that the relation $\frac{\partial}{\partial \boldsymbol{\eta}} - \log g(\boldsymbol{\eta}) = \langle \mathbf{u}(\mathbf{w}) \rangle_{p(\mathbf{w}|\boldsymbol{\eta})}$ holds we take the derivative of the identity $\int p(\mathbf{w}|\boldsymbol{\eta}) d\mathbf{w} - 1 = 0$ with respect to $\boldsymbol{\eta}$, so that

$$\left(\frac{\partial}{\partial \boldsymbol{\eta}} g(\boldsymbol{\eta})\right) \int h(\mathbf{w}) \exp\left(\boldsymbol{\eta}^\mathsf{T} \mathbf{u}(\mathbf{w})\right) d\mathbf{w} = -g(\boldsymbol{\eta}) \int \mathbf{u}(\mathbf{w}) h(\mathbf{w}) \exp\left(\boldsymbol{\eta}^\mathsf{T} \mathbf{u}(\mathbf{w})\right) d\mathbf{w}$$
$$\frac{\partial}{\partial \boldsymbol{\eta}} \log g(\boldsymbol{\eta}) \int g(\boldsymbol{\eta}) h(\mathbf{w}) \exp\left(\boldsymbol{\eta}^\mathsf{T} \mathbf{u}(\mathbf{w})\right) d\mathbf{w} = -\left\langle \mathbf{u}(\mathbf{w}) \right\rangle_{p(\mathbf{w}|\boldsymbol{\eta})}$$

where the last line above follows from making the substitution $\frac{\partial}{\partial \boldsymbol{\eta}} g(\boldsymbol{\eta}) = g(\boldsymbol{\eta}) \frac{\partial}{\partial \boldsymbol{\eta}} \log g(\boldsymbol{\eta})$.

A.5 Potential functions

In this section we provide explicit parametric forms for each of the potential functions considered in this thesis and implemented in the `vgai` package. For each potential we give its moments and the derivative of the log density function which is required to implement the affine independent KL approximate inference method presented in Chapter 6.

A.5.1 Logistic

For a logistic distributed random variable, $v \sim \text{Logistic}(m, s)$, we parameterise its density using

$$\text{Logistic}(v|m, s) := \frac{e^{-r}}{s\left(1 + e^{-r}\right)^2}, \quad \text{where} \quad r := \frac{v - m}{s},$$

with location parameter $m \in \mathbb{R}$ and scale parameter $s \in \mathbb{R}^+$. The logistic density has moments: mean $\langle v \rangle = \mu$, variance $\text{var}(v) = \frac{1}{3}\pi\sigma^2$, skew $\text{skw}(v) = 0$ and excess kurtosis $\text{kur}(v) = \frac{6}{5}$. The derivative of the log density is given by

$$\frac{\partial}{\partial v} \log \text{Logistic}(v|m, s) = -\frac{1}{s} + 2\left(\frac{e^{-r}}{s(1 + e^{-r})}\right).$$

A.5.2 Laplace

For a Laplace distributed random variable, $v \sim \text{Laplace}(m, s)$, we parameterise its density using

$$\text{Laplace}(v|m, s) := \frac{1}{2s} e^{-|r|}, \quad \text{where} \quad r := \frac{v - m}{s},$$

with location parameter $m \in \mathbb{R}$ and scale parameter $s \in \mathbb{R}^+$. The Laplace density has moments: $\langle v \rangle = \mu$, $\text{var}(v) = 2s^2$, $\text{skw}(v) = 0$ and $\text{kur}(v) = 3$. The derivative of the log Laplace density is not defined at its mean/mode. Excluding this point the derivative can be expressed using the $\texttt{sgn} : \mathbb{R} \to \{-1, +1\}$ function which returns the unit sign of its argument, so that

$$\frac{\partial}{\partial v} \log \text{Laplace}(v|m, s) = \frac{-\texttt{sgn}(r)}{s}.$$

Analytic Gaussian expectation of log Laplace potential. The Gaussian expectation of the logarithm of a Laplace potential can be expressed analytically. Laplace potentials, as considered here, take the product of site projections form. Accordingly, to perform G-KL approximate inference, we need only evaluate the derivatives with respect to μ and σ^2. We consider the case of a zero mean Laplace density, $p(\mathbf{w}^\mathsf{T}\mathbf{h}_n|s) = e^{-|\mathbf{w}^\mathsf{T}\mathbf{h}_n|/s}/2s$, giving

$$\left\langle \log p(z|s) \right\rangle_{\mathcal{N}(z|\mu,\sigma^2)} = \left\langle \log p(\mu + z\sigma) \right\rangle_z = -\log(2s) - \frac{1}{s}\left\langle |\mu + z\sigma| \right\rangle_z. \tag{A.5.1}$$

Laplace potentials with non-zero mean, $p(x) = e^{-|x-m|/s}/s$, can be calculated by making the transformation $\mu' = \mu - m$. Evaluating the last term of equation (A.5.1) above involves computing the expectation of a rectified univariate Gaussian random variable,

$$\langle|\mu + z\sigma|\rangle_z = \left(\frac{2}{\pi}\right)^{\frac{1}{2}} \sigma e^{-\frac{1}{2}a_n^2} + \mu\left[1 - 2\Phi\left(-a_n\right)\right]$$

where $\Phi(x) = \int_{-\infty}^{x} \mathcal{N}\left(t|0,1\right) dt$ and $a = \mu/\sigma$. The corresponding derivatives of which are

$$\frac{\partial}{\partial\mu}\langle|\mu + z\sigma|\rangle = 1 - 2\Phi\left(-a\right),$$
$$\frac{\partial}{\partial\sigma^2}\langle|\mu + z\sigma|\rangle = \frac{a^2+1}{\sqrt{2\pi\sigma^2}} e^{-\frac{1}{2}a^2} - \frac{a^2}{\sigma}\mathcal{N}\left(a|0,1\right).$$

A.5.3 Student's t

For a Student's t distributed random variable, $v \sim \text{Student}(\nu, m, s)$, we parameterise its density using

$$\text{Student}(v|\nu, m, s) = \frac{\Gamma(\frac{\nu+1}{2})}{\Gamma(\frac{\nu}{2})\sqrt{\pi\nu s^2}}\left(1 + \frac{r^2}{\nu}\right)^{-\frac{\nu+1}{2}}, \quad \text{where} \quad r := \frac{v-m}{s},$$

with location parameter $m \in \mathbb{R}$, degrees of freedom parameter $\nu \in \mathbb{R}^+$, and scale parameter $s \in \mathbb{R}^+$. The moments of this density are: $\langle v\rangle = \mu$ for $\nu > 1$, $\text{var}(x) = \sigma^2\frac{\nu}{\nu-2}$ for $\nu > 2$, $\text{skw}(v) = 0$ for $\nu > 3$, and $\text{kur}(v) = \frac{6}{\nu-2}$ for $\nu > 4$. The derivative of the log of the Student's t density is given by

$$\frac{\partial}{\partial v}\log\text{Student}(v|\nu, m, s) = \frac{m}{s}\left(\frac{(\nu+1)r}{\nu + r^2}\right).$$

A.5.4 Cauchy

For a Cauchy distributed random variable, $v \sim \text{Cauchy}(m, s)$, we parameterise its density using

$$\text{Cauchy}(v|m, s) = \frac{1}{\pi s\left(1 + r^2\right)}, \quad \text{where} \quad r := \frac{v-m}{s}$$

with location parameter $m \in \mathbb{R}$ and scale $s \in \mathbb{R}^+$. For Cauchy distributed random variables the mean and all higher order moments are undefined for all values of m, s. The derivative of the log Cauchy density is given by

$$\frac{\partial}{\partial v}\log\text{Cauchy}(v|m, s) = \frac{-2r}{s(1 + r^2)}.$$

A.5.5 Sigmoid : logit

The logistic sigmoid distribution for v a binary valued random variable, $v \in \{-1, +1\}$, is parameterised using

$$p(v = +1|m) = \frac{1}{1 + e^{-m}} =: \sigma_{logit}(m),$$

with location parameter $m \in \mathbb{R}$. The logistic sigmoid has the symmetry property that $p(v = +1|m) = 1 - p(v = -1|m)$ so that $p(v|m) = \sigma_{logit}(vm)$ for $v \in \{-1, +1\}$. The derivative of the log of the logistic sigmoid is given by

$$\frac{\partial}{\partial m}\log\sigma_{logit}(m) = 1 - \sigma_{logit}(m).$$

A.5.6 Sigmoid : probit

The probit sigmoid distribution for binary random variables $v \in \{-1, +1\}$ is parameterised using

$$p(v = +1|m) = \int_{-\infty}^{m} \mathcal{N}(x|0, 1)\, dx = \Phi(m) =: \sigma_{probit}(m),$$

with location parameter m. The logistic probit again has the symmetry property that $p(v = +1|m) = 1 - p(v = -1|m)$ so that $p(v|m) = \sigma_{probit}(vm)$ for $y \in \{-1, +1\}$. The derivative of the log sigmoid probit function is given by

$$\frac{\partial}{\partial m} \log \sigma_{probit}(m) = \frac{\mathcal{N}(m)}{\Phi(m)},$$

where $\mathcal{N}(m)$ denotes the standard normal density evaluated at m.

A.5.7 Sigmoid : mixture of Heaviside step functions

As advocated in Hyun-Chul and Ghahramani [2006] a mixture of Heaviside step functions can be used as a noise robust probability mass function for binary classification in latent linear models. The distribution for binary random variables $v \in \{-1, +1\}$ is parameterised using

$$p(v = +1|m, \epsilon) = \begin{cases} \epsilon, & m < 0 \\ 1 - \epsilon, & m \geq 0 \end{cases} = (1 - 2\epsilon)\mathbb{I}[m > 0] + \epsilon =: \sigma_{heavi}(m)$$

where $\epsilon \in [0, \frac{1}{2})$ is a parameter specifying the label misclassification rate or noise and $m \in \mathbb{R}$ is a location parameter. For this distribution the symmetry property holds such that $p(v = -1|m, \epsilon) = 1 - p(v = 1|m, \epsilon)$ and so $p(v|m, \epsilon) = \sigma_{heavi}(vm)$. The derivative of the mixture heaviside sigmoid is zero for all $m \neq 0$ and can be represented by the Dirac delta when we take its expectation

$$\frac{\partial}{\partial m} \log \sigma_{heavi}(m) = \log\left(\frac{1 - \epsilon}{\epsilon}\right) \delta(m).$$

Analytic Gaussian expectation of log Heaviside mixture sigmoid. The univariate Gaussian expectation of the log sigmoid heaviside potential has a simple analytic expression. Below we present these expectations, and their corresponding derivatives, for efficient evaluation of the G-KL bound.

$$\langle \log \sigma_{heavi}(z) \rangle_{\mathcal{N}(z|\mu,\sigma^2)} = \log\left(\frac{\epsilon}{1 - \epsilon}\right) \Phi\left(-\frac{\mu}{\sigma}\right) + \log(1 - \epsilon)$$

where $\Phi(z) := \int_{-\infty}^{z} \mathcal{N}(z|0, 1)$. Which admits the gradients

$$\frac{\partial}{\partial \mu} \langle \log \sigma_{heavi}(z) \rangle_{\mathcal{N}(z|\mu,\sigma^2)} = -\log\left(\frac{1 - \epsilon}{\epsilon}\right) \mathcal{N}\left(\frac{\mu}{\sigma}\right) \frac{1}{\sigma},$$
$$\frac{\partial}{\partial \sigma^2} \langle \log \sigma_{heavi}(z) \rangle_{\mathcal{N}(z|\mu,\sigma^2)} = -\frac{1}{2} \log\left(\frac{1 - \epsilon}{\epsilon}\right) \mathcal{N}\left(\frac{\mu}{\sigma}\right) \frac{\mu}{\sigma^3}.$$

A.6 Matrix identities

Below we specify some of the core linear algebra matrix identities that are used throughout the thesis. The results in this section are taken from Boyd and Vandenberghe [2004], Golub and Van Loan [1996].

A.6.1 Cholesky factorisation

If $\mathbf{S} \in \mathbb{R}^{D\times D}$ is a symmetric positive definite matrix then it can be uniquely factorised as $\mathbf{S} = \mathbf{C}^\mathsf{T}\mathbf{C}$ where $\mathbf{C} \in \mathbb{R}^{D\times D}$ is an upper-triangular non-singular matrix with positive diagonals. $\mathbf{C}$ is called the Cholesky factorisation of $\mathbf{S}$. Computing the Cholesky factorisation scales $O\left(\frac{1}{3}D^3\right)$ and is generally a very numerically stable procedure. Given the Cholesky factorisation of a symmetric positive definite matrix $\mathbf{S}$, various computations involving $\mathbf{S}$ can be performed at a reduced complexity than working with $\mathbf{S}$ directly. Some of these techniques are:

- The Cholesky factorisation is the preferred method of solving the linear system $\mathbf{S}\mathbf{x} = \mathbf{b}$, since $\mathbf{x} = \mathbf{C}^{-1}\mathbf{C}^{-\mathsf{T}}\mathbf{b}$ and since $\mathbf{C}^{-1}$ is triangular, $\mathbf{x}$ can be evaluated by two back substitutions and so scales $O\left(2D^2\right)$.
- Once the Cholesky factorisation of $\mathbf{S}$ has been computed, the determinant $\det(\mathbf{S})$, and the log determinant $\log\det(\mathbf{S})$, can be evaluated in $O(D)$ time since $\det(\mathbf{C}) = \prod_d C_{dd}$ and so $\det(\mathbf{S}) = \det(\mathbf{C})^2 = \prod_d C_{dd}^2$.
- Efficient routines exist to perform rank one updates of Cholesky factorisations. Defining $\mathbf{S}' := \mathbf{S} + \mathbf{x}\mathbf{x}^\mathsf{T}$, where we already have $\mathbf{C}$ such that $\mathbf{S} = \mathbf{C}^\mathsf{T}\mathbf{C}$, then $\mathbf{C}'$ the Cholesky factorisation of $\mathbf{S}'$ can be computed in $O\left(D^2\right)$ time [Seeger, 2007].

A.6.2 LU factorisation

Every non-singular matrix $\mathbf{A} \in \mathbb{R}^{D\times D}$ can be factorised as $\mathbf{A} = \mathbf{P}\mathbf{L}\mathbf{U}$, where $\mathbf{P} \in \mathbb{R}^{D\times D}$ is a permutation matrix, $\mathbf{L} \in \mathbb{R}^{D\times D}$ is a lower triangular matrix and $\mathbf{U} \in \mathbb{R}^{D\times D}$ is a upper-triangular matrix. Such a factorisation is referred to as the LU factorisation. For general unstructured $\mathbf{A}$ computing the LU factorisation scales $O\left(\frac{2}{3}D^3\right)$. Similarly to the Cholesky factorisation the LU factorisation can be used to make computations with respect to $\mathbf{A}$ cheaper, some of these methods include:

- Solving the linear system $\mathbf{A}\mathbf{x} = \mathbf{b}$ can be performed by sequential back substitutions: solve $\mathbf{P}\mathbf{z}_1 = \mathbf{b}$ using $\mathbf{z}_1 = \mathbf{P}^\mathsf{T}\mathbf{b}$, then solve $\mathbf{L}\mathbf{z}_2 = \mathbf{z}_1$ by forward substitution, then solve $\mathbf{U}\mathbf{x} = \mathbf{z}_2$ by back substitution. Thus, provided with the LU factorisation of $\mathbf{A}$, solving the linear system $\mathbf{A}^{-1}\mathbf{x} = \mathbf{b}$ scales $O\left(2D^2\right)$.
- The determinant of $\mathbf{A}$ can be computed in $O(2D)$ time since $\det(\mathbf{A}) = (-1)^{\#\text{row}}\det(\mathbf{L})\det(\mathbf{U}) = (-1)^{\#\text{row}}\prod_d L_{dd}U_{dd}$, where $\#$row denotes the number of row permutations defined by the permutation $\mathbf{P}$.

A.6.3 Matrix inversion lemma

The matrix inversion lemma, otherwise known as the Sherman-Morrison-Woodbury identity, provides a means to potentially compute the inverse and determinant of a structured square matrix more efficiently than direct evaluation. For square matrices $\mathbf{A} \in \mathbb{R}^{D\times D}$ and $\mathbf{\Gamma} \in \mathbb{R}^{N\times N}$ and matrices $\mathbf{U}, \mathbf{V}^\mathsf{T} \in \mathbb{R}^{D\times N}$ then

$$(\mathbf{A} + \mathbf{U}\mathbf{\Gamma}\mathbf{V})^{-1} = \mathbf{A}^{-1} - \mathbf{A}^{-1}\mathbf{U}\left(\mathbf{\Gamma}^{-1} + \mathbf{V}\mathbf{A}^{-1}\mathbf{U}\right)^{-1}\mathbf{V}\mathbf{A}^{-1}. \tag{A.6.1}$$

Provided that $N < D$ and $\mathbf{A}^{-1}$ can be efficiently computed, for example it is diagonal or banded, this identity provides a possibly more efficient means to compute the inverse as expressed on the left hand side of equation (A.6.1).

Matrix determinant lemma. This identity expressed in equation (A.6.1) can also by used to evaluate the determinant of a matrix that satisfies the same factorisation structure. Specifically we have that

$$\det\left(\mathbf{A}+\mathbf{U}\boldsymbol{\Gamma}\mathbf{V}\right)=\det\left(\mathbf{A}\right)\det\left(\boldsymbol{\Gamma}\right)\det\left(\boldsymbol{\Gamma}^{-1}+\mathbf{V}\mathbf{A}^{-1}\mathbf{U}\right). \tag{A.6.2}$$

A.7 Deterministic approximation inference

A.7.1 Mean field equations

Following Csató et al. [2000], for a factorising approximation $q(\mathbf{w})=\prod_{d=1}^{D}q(w_d)$ we derive the mean field updates for a target density of the form

$$p(\mathbf{w})=\frac{1}{Z}\mathcal{N}\left(\mathbf{w}|\boldsymbol{\mu},\boldsymbol{\Sigma}\right)\prod_{d=1}^{D}\phi_d(w_d).$$

Importantly the non-Gaussian potential factorises over the dimensions of $\mathbf{w}$. The KL variational bound for a target of this form and a the fully factorising approximation $q(\mathbf{w})$ then takes the form

$$\log Z\geq\mathcal{B}_{MF}:=\sum_{d=1}^{D}-\left\langle\log q(w_d)\right\rangle_{q(w_d)}+\left\langle\log\mathcal{N}\left(\mathbf{w}|\boldsymbol{\mu},\boldsymbol{\Sigma}\right)\right\rangle_{\prod_d q(w_d)}+\sum_{d=1}^{D}\left\langle\log\phi_d(w_d)\right\rangle_{q(w_d)}.$$

Considering the Gaussian potential's contribution to the bound first

$$2\left\langle\log\mathcal{N}\left(\mathbf{w}|\boldsymbol{\mu},\boldsymbol{\Sigma}\right)\right\rangle_{\prod_d q(w_d)}=-\log\det\left(2\pi\boldsymbol{\Sigma}\right)-\boldsymbol{\mu}^{\mathsf{T}}\boldsymbol{\Sigma}^{-1}\boldsymbol{\mu}-\left\langle\mathbf{w}^{\mathsf{T}}\boldsymbol{\Sigma}^{-1}\mathbf{w}\right\rangle+2\left\langle\mathbf{w}\right\rangle\boldsymbol{\Sigma}^{-1}\boldsymbol{\mu}.$$

We let $\boldsymbol{\Lambda}=\boldsymbol{\Sigma}^{-1}$ to ease notation,

$$\begin{aligned}\left\langle\mathbf{w}^{\mathsf{T}}\boldsymbol{\Sigma}^{-1}\mathbf{w}\right\rangle&=\sum_i\int q(w_i)\Lambda_{ii}w_i^2dw_i+\int q(\mathbf{w})\sum_{i,j:i\neq j}w_iw_j\Lambda_{ij}d\mathbf{w}\\&=\sum_i\int w_i^2q(w_i)\Lambda_{ii}dw_i+\sum_i\int w_iq(w_i)\sum_{j\neq i}\int q(w_j)w_j\Lambda_{ij}dw_idw_j\end{aligned}$$

The functional derivative of this term with respect to $q(w_k)$ can be written as

$$\frac{\partial}{\partial q(w_k)}\left\langle\mathbf{w}^{\mathsf{T}}\boldsymbol{\Sigma}^{-1}\mathbf{w}\right\rangle=\Lambda_{kk}w_k^2+w_k\sum_{j\neq k}\int w_jq(w_j)\Lambda_{kj}dw_j.$$

Taking the functional derivative of the bound as a whole we get

$$\frac{\partial}{\partial q(w_k)}\mathcal{B}_{MF}=\log q(w_k)+\log\phi_k(w_k)+w_k\left[\boldsymbol{\Lambda}\boldsymbol{\mu}\right]_k-\frac{1}{2}\Lambda_{kk}w_k^2-w_k\sum_{j\neq k}\left\langle w_j\right\rangle\Lambda_{kj}.$$

Equating the derivative above to zero and exponentiating we get

$$q(w_k)\propto\phi_k(w_k)\exp\left(-w_ka_k+\frac{1}{2}\Lambda_{kk}w_k^2\right),\qquad\text{where}\qquad a_k:=-\left[\boldsymbol{\Lambda}\boldsymbol{\mu}\right]_k+\sum_{j\neq k}\left\langle w_j\right\rangle\Lambda_{kj},$$

which can be expressed as the following product of the site potential and a Gaussian

$$q(w_k)=\frac{1}{Z_k}\phi_k(w_k)\mathcal{N}\left(w_k\left|\frac{a_k}{\Lambda_{kk}},\Lambda_{kk}^{-1}\right.\right). \tag{A.7.1}$$

Optimising the mean field bound $\mathcal{B}_{MF}$ requires asynchronously updating each of the factors of the factorising density as defined in equation (A.7.1). Whether the integrals required to define the moments and the normalisation constant can be analytically computed depends on the analytic form of the potential functions ϕ_k considered. We note that since all integrals are univariate they can be computed cheaply using some univariate numerical integration procedure. In what follows we denote the moments $m_k := \int w_k q(w_k) dw_k$ which is the k^{th} element of the vector $\mathbf{m}$ and $s_k := \int (w_k - m_k)^2 q(w_k) dw_k$ which is the k^{th} element of the the vector $\mathbf{s}$.

Plugging the optimised factorising approximation $q(\mathbf{w})$ defined by the factors in equation (A.7.1) into the bound we get

$$\mathcal{B}_{MF} = \sum_d H[q(w_k)] - \frac{1}{2}\log\det(2\pi\boldsymbol{\Sigma}) - \frac{1}{2}\boldsymbol{\mu}^\mathsf{T}\boldsymbol{\Sigma}^{-1}\boldsymbol{\mu} - \frac{1}{2}\mathbf{s}^\mathsf{T}\mathrm{diag}\left(\boldsymbol{\Sigma}^{-1}\right) - \frac{1}{2}\mathbf{m}^\mathsf{T}\boldsymbol{\Sigma}^{-1}\mathbf{m} + \sum_d \left\langle \log \phi_d(w_d) \right\rangle_{q(w_d)},$$

where the $\mathrm{diag}(\cdot)$ operator constructs a column vector from the diagonal elements of a square matrix or a diagonal matrix from a column vector.

Appendix B

Gaussian KL approximate inference

In this appendix we present additional results concerning Gaussian Kullback-Liebler approximate inference methods that were the subject of Chapters 4 and 5. In Appendix B.1 we present various identities required to efficiently evaluate and optimise the G-KL bound by gradient ascent. In Appendix B.2 we discuss in greater depth the constrained subspace parameterisation of G-KL covariance. In Appendix B.3 we provide conditions for which G-KL bound optimisation will exhibit quadratic convergence rates using Newton's method. In Appendix B.4 we present the computational complexity scaling figures of G-KL bound and derivative evaluations for a range of potentials and parameterisations of covariance. In Appendix B.5 we present a technique that can be used to reduce the complexity of inference problems where the Gaussian potential has an unstructured covariance matrix. In Appendix B.6 we present various details required to apply G-KL methods to Gaussian process models. In Appendix B.7 we present an alternative derivation of the G-KL bound concavity result as originally published in Challis and Barber [2011]. Finally in Appendix B.8 we present documentation for the G-KL approximate inference Matlab package `vgai`.

B.1 G-KL bound and gradients

We present the G-KL bound and its gradient for Gaussian and generic site projection potentials with full Cholesky and factor analysis parameterisations of G-KL covariance. Gradients for the chevron, banded and sparse Cholesky covariance parameterisations are implemented simply by placing that Cholesky parameterisation's sparsity mask on the full Cholesky gradient matrix. Subspace Cholesky G-KL gradients and the associated optimisation procedures are discussed in Section B.2.

B.1.1 Entropy

For the Cholesky decomposition of covariance, $\mathbf{S} = \mathbf{C}^\mathsf{T}\mathbf{C}$, the entropy term of the G-KL bound and its gradient with respect to $\mathbf{C}$ are given by

$$
\begin{aligned}
-\left\langle \log q(\mathbf{w}) \right\rangle_{q(\mathbf{w})} &= \frac{D}{2}\log(2\pi) + \frac{D}{2} + \sum_{d=1}^{D} \log(C_{dd}), \\
\frac{\partial}{\partial C_{ij}} -\left\langle \log q(\mathbf{w}) \right\rangle_{q(\mathbf{w})} &= \delta_{ij}\frac{1}{C_{ij}},
\end{aligned}
$$

where δ_{ij} is the Kronecker delta. For the factor analysis (FA) parameterisation of G-KL covariance, $\mathbf{S} = \mathrm{diag}\left(\mathbf{d}^2\right) + \boldsymbol{\Theta}\boldsymbol{\Theta}^\mathsf{T}$ where $\mathbf{d} \in \mathbb{R}^D$ and $\boldsymbol{\Theta} \in \mathbb{R}^{D\times K}$, the entropy is given by,

$$-\left\langle \log q(\mathbf{w}) \right\rangle = \frac{D}{2}\log(2\pi) + \frac{D}{2} + \sum_d \log(d_d) + \frac{1}{2}\log\det\left(\mathbf{I}_{K\times K} + \boldsymbol{\Theta}^\mathsf{T}\mathrm{diag}\left(\frac{1}{\mathbf{d}^2}\right)\boldsymbol{\Theta}\right),$$

admitting the gradients,

$$\frac{\partial}{\partial \mathbf{d}}\left\langle \log q(\mathbf{w}) \right\rangle_{q(\mathbf{w})} = 2\mathbf{d} \odot \mathrm{diag}\left(\mathbf{S}^{-1}\right),$$
$$\frac{\partial}{\partial \boldsymbol{\Theta}}\left\langle \log q(\mathbf{w}) \right\rangle_{q(\mathbf{w})} = 2\mathbf{S}^{-1}\boldsymbol{\Theta}.$$

Where $\odot$ refers to taking the element wise product and $\mathrm{diag}\left(\right)$ refers to either constructing a square diagonal matrix from a column vector or forming a column vector from the diagonal elements of a square matrix. Evaluating $\mathbf{S}^{-1}$ scales $O\left(K^2D\right)$ using the Woodbury matrix inversion identity:

$$\mathbf{S}^{-1} = \mathrm{diag}\left(\frac{1}{\mathbf{d}^2}\right) - \mathrm{diag}\left(\frac{1}{\mathbf{d}^2}\right)\boldsymbol{\Theta}\left(\mathbf{I}_{K\times K} + \boldsymbol{\Theta}^\mathsf{T}\mathrm{diag}\left(\frac{1}{\mathbf{d}^2}\right)\boldsymbol{\Theta}\right)^{-1}\boldsymbol{\Theta}^\mathsf{T}\mathrm{diag}\left(\frac{1}{\mathbf{d}^2}\right).$$

B.1.2 Site projection potentials

Each site projection potential's contribution to the G-KL bound can be expressed as

$$I_n := \left\langle \log\phi_n(\mathbf{w}^\mathsf{T}\mathbf{h}_n)\right\rangle = \left\langle \log\phi_n(y)\right\rangle_{\mathcal{N}(y|m_n,s_n^2)} = \left\langle \log\phi_n(m_n + zs_n)\right\rangle_{\mathcal{N}(z|0,1)},$$

where $m_n = \mathbf{h}_n^\mathsf{T}\mathbf{m}$ and $s_n^2 = \mathbf{h}_n^\mathsf{T}\mathbf{S}\mathbf{h}_n$. In order that general potentials of this form can be easily implemented for different functions ϕ_n we present the gradients according to their chain rule decomposition,

$$\frac{\partial I_n}{\partial \mathbf{m}} = \frac{\partial I_n}{\partial m_n}\frac{\partial m_n}{\partial \mathbf{m}} \quad \text{and} \quad \frac{\partial I_n}{\partial \mathbf{C}} = \frac{\partial I_n}{\partial s_n^2}\frac{\partial s_n^2}{\partial \mathbf{C}}. \tag{B.1.1}$$

Expressing I_n and its derivative as an expectation with respect to the standard normal density renders the implementation of numerical integration routines simpler and avoids having to derive the potential functions derivative. Doing so, I_n and its derivatives are given by:

$$I_n = \int \mathcal{N}(z|0,1)\log\phi_n(m_n + zs_n)dz,$$
$$\frac{\partial I_n}{\partial m_n} = \int z\mathcal{N}(z|0,1)\frac{\log\phi_n(m_n + zs_n)}{s_n}dz,$$
$$\frac{\partial I_n}{\partial s_n^2} = \int \left(z^2 - 1\right)\mathcal{N}(z|0,1)\frac{\log\phi_n(m_n + zs_n)}{2s_n^2}dz.$$

The partial derivatives of $m_n = \mathbf{h}_n^\mathsf{T}\mathbf{m}$ and $s_n^2 = \mathbf{h}_n^\mathsf{T}\mathbf{S}\mathbf{h}_n$ are

$$\frac{\partial m_n}{\partial \mathbf{m}} = \mathbf{h}_n, \quad \text{and} \quad \frac{\partial s_n^2}{\partial \mathbf{C}} = 2\mathrm{triu}\left(\mathbf{C}\mathbf{h}_n\mathbf{h}_n^\mathsf{T}\right),$$

where $\mathrm{triu}\left(\cdot\right)$ is a sparsity mask such that elements below the diagonal are fixed to zero. For FA parameterisations we have

$$\frac{\partial s_n^2}{\partial \mathbf{d}} = 2\mathbf{h}_n^2 \odot \mathbf{d}, \quad \text{and} \quad \frac{\partial s_n^2}{\partial \boldsymbol{\Theta}} = 2\mathbf{h}_n\mathbf{h}_n^\mathsf{T}\boldsymbol{\Theta}.$$

B.1.3 Gaussian potentials

For a Gaussian potential $\mathcal{N}(\mathbf{w}|\boldsymbol{\mu}, \boldsymbol{\Sigma})$ the log expectation is given by

$$\langle \log \mathcal{N}(\mathbf{w}|\boldsymbol{\mu}, \boldsymbol{\Sigma}) \rangle_{q(\mathbf{w})} = -\frac{1}{2}\left[\log\det(2\pi\boldsymbol{\Sigma}) + (\mathbf{m}-\boldsymbol{\mu})^{\mathsf{T}}\boldsymbol{\Sigma}^{-1}(\mathbf{m}-\boldsymbol{\mu}) + \mathrm{trace}\left(\boldsymbol{\Sigma}^{-1}\mathbf{S}\right)\right].$$

Derivatives with respect to the mean and covariance are

$$\frac{\partial}{\partial \mathbf{m}}\langle \log \mathcal{N}(\mathbf{w}|\boldsymbol{\mu}, \boldsymbol{\Sigma})\rangle = \boldsymbol{\Sigma}^{-1}(\boldsymbol{\mu}-\mathbf{m}), \quad \text{and} \quad \frac{\partial}{\partial \mathbf{C}}\langle \log \mathcal{N}(\mathbf{w}|\boldsymbol{\mu}, \boldsymbol{\Sigma})\rangle = -\mathrm{triu}\left(\mathbf{C}\boldsymbol{\Sigma}^{-1}\right).$$

For the FA covariance structure we have,

$$\frac{\partial}{\partial \mathbf{d}}\langle \log \mathcal{N}(\mathbf{w}|\boldsymbol{\mu}, \boldsymbol{\Sigma})\rangle = -\mathrm{diag}\left(\boldsymbol{\Sigma}^{-1}\right)\odot \mathbf{d}, \quad \text{and} \quad \frac{\partial}{\partial \boldsymbol{\Theta}}\langle \log \mathcal{N}(\mathbf{w}|\boldsymbol{\mu}, \boldsymbol{\Sigma})\rangle = -\boldsymbol{\Sigma}^{-1}\boldsymbol{\Theta}.$$

Gaussian likelihoods

Linear models with additive Gaussian noise have a likelihood potential that can be expressed as $\mathcal{N}\left(\mathbf{y}|\mathbf{H}^{\mathsf{T}}\mathbf{w}, \boldsymbol{\Sigma}\right)$ where $\mathbf{H} \in \mathbb{R}^{D\times N}$ and $\mathbf{y} \in \mathbb{R}^{N}$. In this setting typically we assume isotropic noise $\boldsymbol{\Sigma} = \nu^2\mathbf{I}$ and so present gradients for this case only. The expectation of the log of this term has the following algebraic from

$$\left\langle \log \mathcal{N}\left(\mathbf{y}|\mathbf{H}^{\mathsf{T}}\mathbf{w}, \nu^2\mathbf{I}\right)\right\rangle = -\frac{1}{2}\left[N\log(2\pi\nu^2) + \frac{1}{\nu^2}\left\langle \left(\mathbf{y}-\mathbf{H}^{\mathsf{T}}\mathbf{w}\right)^{\mathsf{T}}\left(\mathbf{y}-\mathbf{H}^{\mathsf{T}}\mathbf{w}\right)\right\rangle\right], \quad \text{(B.1.2)}$$

where the expectation of the quadratic can be expressed as

$$\left\langle \left(\mathbf{y}-\mathbf{H}^{\mathsf{T}}\mathbf{w}\right)^{\mathsf{T}}\left(\mathbf{y}-\mathbf{H}^{\mathsf{T}}\mathbf{w}\right)\right\rangle = \mathbf{y}^{\mathsf{T}}\mathbf{y} - 2\mathbf{y}^{\mathsf{T}}\mathbf{H}^{\mathsf{T}}\mathbf{m} + \sum_{ij}[\mathbf{C}\mathbf{H}]_{ij}^2 + \sum_i\left[\mathbf{H}^{\mathsf{T}}\mathbf{m}\right]_i^2.$$

Equation (B.1.2) admits the gradients:

$$\frac{\partial}{\partial \mathbf{m}}\left\langle \log \mathcal{N}\left(\mathbf{y}|\mathbf{H}^{\mathsf{T}}\mathbf{w}, \nu^2\mathbf{I}\right)\right\rangle = \frac{1}{\nu^2}\left(\mathbf{y}^{\mathsf{T}}\mathbf{H}^{\mathsf{T}} - \mathbf{H}\mathbf{H}^{\mathsf{T}}\mathbf{m}\right),$$
$$\frac{\partial}{\partial \mathbf{C}}\left\langle \log \mathcal{N}\left(\mathbf{y}|\mathbf{H}^{\mathsf{T}}\mathbf{w}, \nu^2\mathbf{I}\right)\right\rangle = -\frac{1}{\nu^2}\mathrm{triu}\left(\mathbf{C}\mathbf{H}\mathbf{H}^{\mathsf{T}}\right).$$

For the FA parameterised covariance we have

$$\left\langle \left(\mathbf{y}-\mathbf{H}^{\mathsf{T}}\mathbf{w}\right)^{\mathsf{T}}\left(\mathbf{y}-\mathbf{H}^{\mathsf{T}}\mathbf{w}\right)\right\rangle = \mathbf{y}^{\mathsf{T}}\mathbf{y} - 2\mathbf{y}^{\mathsf{T}}\mathbf{H}^{\mathsf{T}}\mathbf{m} + \sum_i\left[\mathbf{H}^{\mathsf{T}}\mathbf{m}\right]_i^2 + \sum_{ij}\left[\boldsymbol{\Theta}^{\mathsf{T}}\mathbf{H}^{\mathsf{T}}\right]_{ij}^2 + \sum_j\left(\sum_i H_{ji}^2\right)d_j^2$$

with corresponding gradients:

$$\frac{\partial}{\partial d_j}\left\langle \log \mathcal{N}\left(\mathbf{y}|\mathbf{H}^{\mathsf{T}}\mathbf{w}, \nu^2\mathbf{I}\right)\right\rangle = -\frac{1}{\nu^2}\left(\sum_i H_{ji}^2\right)d_j,$$
$$\frac{\partial}{\partial \boldsymbol{\Theta}}\left\langle \log \mathcal{N}\left(\mathbf{y}|\mathbf{M}\mathbf{w}, \nu^2\mathbf{I}\right)\right\rangle = -\frac{1}{\nu^2}\mathbf{H}\mathbf{H}^{\mathsf{T}}\boldsymbol{\Theta}.$$

Gaussian potentials as site projections

The Gaussian potential $\mathcal{N}(\mathbf{w}|\boldsymbol{\mu}, \boldsymbol{\Sigma})$ can be equivalently expressed as a product of D site projection potentials. To see this we use the Cholesky factorisation of the precision matrix $\boldsymbol{\Sigma}^{-1} = \mathbf{P}^{\mathsf{T}}\mathbf{P}$. Making

this substitution, we see that

$$\mathcal{N}\left(\mathbf{w}|\boldsymbol{\mu},\boldsymbol{\Sigma}\right) \propto e^{-\frac{1}{2}(\mathbf{w}-\boldsymbol{\mu})^{\mathsf{T}}\mathbf{P}^{\mathsf{T}}\mathbf{P}(\mathbf{w}-\boldsymbol{\mu})} = e^{-\frac{1}{2}\|\mathbf{P}(\mathbf{w}-\boldsymbol{\mu})\|_2^2} = \prod_{d=1}^{D} e^{-\frac{1}{2}\left(\mathbf{p}_d^{\mathsf{T}}(\mathbf{w}-\boldsymbol{\mu})\right)^2}, \tag{B.1.3}$$

where the vector $\mathbf{p}_d$ is the d^{th} row vector of $\mathbf{P}$, that is $\mathbf{p}_d := \mathbf{P}_{d,:}$. Thus Equation B.1.3 is a product of D site projections with potential function $\phi_d(x) \propto e^{-\frac{1}{2}x^2}$.

B.2 Subspace covariance decomposition

We consider optimising the G-KL bound with respect to a covariance matrix parameterised on a subspace of the parameters $\mathbf{w} \in \mathbb{R}^D$. Letting $\mathbf{E} = [\mathbf{E}_1, \mathbf{E}_2]$ be a matrix of orthonormal vectors that span $\mathbb{R}^D$ such that $\mathbf{E}_1^{\mathsf{T}}\mathbf{E}_1 = \mathbf{I}_{K\times K}$ and $\mathbf{E}_2^{\mathsf{T}}\mathbf{E}_2 = \mathbf{I}_{L\times L}$ where $L := D - K$, then we may parameterise the covariance as

$$\mathbf{S}' = \mathbf{E}\mathbf{S}\mathbf{E}^{\mathsf{T}} = [\mathbf{E}_1, \mathbf{E}_2]\,\mathbf{S}\,[\mathbf{E}_1, \mathbf{E}_2]^{\mathsf{T}},$$

which is equivalent to making an orthonormal transformation in the space of parameters $\mathbf{w}$ using $\mathbf{E}$. If we restrict $\mathbf{S}$ to be block diagonal, $\mathbf{S} = \mathrm{blkdiag}\left(\mathbf{S}_1, \mathbf{S}_2\right)$ where $\mathbf{S}_1$ is K-dimensional and $\mathbf{S}_2$ is L dimensional, we can write $\mathbf{S}'$ as the sum

$$\mathbf{S}' = \mathbf{E}_1\mathbf{S}_1\mathbf{E}_1^{\mathsf{T}} + \mathbf{E}_2\mathbf{S}_2\mathbf{E}_2^{\mathsf{T}}.$$

Since $\mathbf{E}$ is orthonormal it does not effect the value or gradient of the entropy's contribution to the bound since $\log\det\left(\mathbf{S}\right) = \log\det\left(\mathbf{S}'\right)$. Provided the Gaussian potential has spherical covariance, $\boldsymbol{\Sigma} = \nu^2\mathbf{I}$, then $\mathbf{E}$ does not effect its contribution the G-KL bound since

$$\mathrm{trace}\left(\boldsymbol{\Sigma}^{-1}\mathbf{S}'\right) = \frac{1}{\nu^2}\mathrm{trace}\left(\mathbf{E}\mathbf{S}\mathbf{E}^{\mathsf{T}}\right) = \frac{1}{\nu^2}\mathrm{trace}\left(\mathbf{S}\right).$$

Thus we are left to evaluate the projected variance terms $\left\{s_n^2\right\}_{n=1}^N$ required to evaluate the product of site potentials contribution. For $\mathbf{S}$ block diagonal with the second block component spherical, $\mathbf{S}_2 = c^2\mathbf{I}$, the orthonormal basis vectors $\mathbf{E}_2$ do not need to be computed or maintained since

$$s_n^2 = \mathbf{h}_n^{\mathsf{T}}\mathbf{S}'\mathbf{h}_n = \mathbf{h}_n^{\mathsf{T}}\mathbf{E}_1\mathbf{S}_1\mathbf{E}_1^{\mathsf{T}}\mathbf{h}_n + c^2\mathbf{h}_n^{\mathsf{T}}\mathbf{E}_2\mathbf{E}_2^{\mathsf{T}}\mathbf{h}_n = \mathbf{h}_n^{\mathsf{T}}\mathbf{E}_1\mathbf{S}_1\mathbf{E}_1^{\mathsf{T}}\mathbf{h}_n + c^2\left(\|\mathbf{h}_n\|_2^2 - \|\mathbf{E}_1^{\mathsf{T}}\mathbf{h}_n\|^2\right).$$

We seek to optimise the G-KL bound with respect to the subspace parameterised variational Gaussian by iterating between optimising the bound with respect to the parameters $\{\mathbf{m}, \mathbf{C}_1, c\}$ and updating the subspace basis vectors $\mathbf{E}_1$. In Section B.2.1 we present the gradients required to optimise the G-KL bound with respect to $\{\mathbf{m}, \mathbf{C}_1, c\}$. In Section B.2.2 and B.2.3 we present two different methods to optimise the subspace basis $\mathbf{E}_1$.

B.2.1 Subspace Cholesky G-KL bound gradients

In this subsection we present the subspace Cholesky G-KL bound gradients. The subspace covariance matrix is given by $\mathbf{S} = \mathbf{E}_1\mathbf{C}_1^{\mathsf{T}}\mathbf{C}_1\mathbf{E}_1^{\mathsf{T}} + c^2\mathbf{E}_2\mathbf{E}_2^{\mathsf{T}}$, where $\mathbf{C}_1 \in \mathbb{R}^{K\times K}$ is a Cholesky matrix, $c \in \mathbb{R}^+$ and $D = K + L$. Since $\mathbf{E}_2$ does not occur in the expressions presented below, in what follows we omit subscripts and denote $\mathbf{E}_1$ and $\mathbf{C}_1$ as $\mathbf{E}$ and $\mathbf{C}$. We reiterate that we assumer here that the Gaussian

potential has spherical covariance $\boldsymbol{\Sigma} = \nu^2\mathbf{I}$. The G-KL bound for the subspace Cholesky covariance parameterisation is given by

$$\begin{aligned}\mathcal{B}_{G\text{-}KL}(\mathbf{m}, \mathbf{C}, c, \mathbf{E}) = \frac{D}{2}\log(2\pi) + \frac{D}{2} + \sum_{k=1}^{K}\log(C_{kk}) + L\log(c) \\ - \frac{D}{2}\log\left(2\pi\nu^2\right) - \frac{1}{\nu^2}\left[\|\mathbf{m}-\boldsymbol{\mu}\|_2^2 + \operatorname{trace}\left(\mathbf{C}^\mathsf{T}\mathbf{C}\right) + Lc^2\right] \\ + \sum_{n=1}^{N}\left\langle\log\phi_n(m_n + zs_n)\right\rangle_{\mathcal{N}(z|0,1)}.\end{aligned}$$

The gradient of the G-KL entropy's contribution to the bound is

$$\frac{\partial}{\partial C_{ij}} - \langle\log q(\mathbf{w})\rangle = \delta_{ij}\frac{1}{C_{ij}}, \quad \text{and} \quad \frac{\partial}{\partial c} - \langle\log q(\mathbf{w})\rangle = \frac{L}{c}.$$

The Gaussian potential's contribution to the G-KL bound admits the gradients:

$$\frac{\partial}{\partial\mathbf{C}}\left\langle\log\mathcal{N}\left(\mathbf{w}|\boldsymbol{\mu},\nu^2\mathbf{I}\right)\right\rangle = -\frac{1}{\nu^2}\mathbf{C}, \quad \text{and} \quad \frac{\partial}{\partial c}\left\langle\log\mathcal{N}\left(\mathbf{w}|\boldsymbol{\mu},\nu^2\mathbf{I}\right)\right\rangle = -\frac{Lc}{\nu^2}.$$

The site projection potential's contribution to the G-KL bound is computed as in Section B.1.2 but with the partial derivatives of s_n^2 with respect to $\mathbf{C}$ and c:

$$\frac{\partial s_n^2}{\partial\mathbf{C}} = 2\operatorname{triu}\left(\mathbf{C}\tilde{\mathbf{h}}_n\tilde{\mathbf{h}}_n^\mathsf{T}\right), \quad \frac{\partial s_n^2}{\partial c} = 2c\left(\|\mathbf{h}_n\|_2^2 - \|\tilde{\mathbf{h}}_n\|_2^2\right),$$

where $\tilde{\mathbf{h}}_n := \mathbf{E}^\mathsf{T}\mathbf{h}_n$.

B.2.2 Subspace optimisation : projected gradient ascent

One route to finding good subspace vectors $\mathbf{E}_1$ is to directly optimise the bound with respect to them. Again we omit subscripts since $\mathbf{E}_2$ makes no contribution to the expressions below. Optimisation is complicated by the fact that we require $\mathbf{E}$ to be orthonormal, *i.e.* we require that $\mathbf{E}^\mathsf{T}\mathbf{E} = \mathbf{I}_{K\times K}$. The set of all such orthonormal vectors forms a smooth manifold in $\mathbb{R}^{D\times K}$. A crude but simple approach to optimising the bound with respect to $\mathbf{E}$ is projected gradient ascent – after each gradient step we orthonormalise the updated basis:

$$\mathbf{E}^{new} := \operatorname{orth}\left[\mathbf{E} + \alpha\frac{\partial}{\partial\mathbf{E}}\mathcal{B}_{G\text{-}KL}(\mathbf{m}, \mathbf{C}, \mathbf{E}, c)\right]$$

where $\operatorname{orth}[\cdot]$ denotes an orthonormalisation operator, implemented for instance using a Gram-Schmidt procedure or the singular value decomposition, and α is a parameter controlling the gradient step size.

As described above, when $\boldsymbol{\Sigma} = \nu^2\mathbf{I}$, the only term in the G-KL bound that depends on $\mathbf{E}$ are the site projection potential functions $\left\langle\log\phi_n(\mathbf{w}^\mathsf{T}\mathbf{h}_n)\right\rangle$. The derivative of the bound then with respect to $\mathbf{E}$ is given by

$$\frac{\partial}{\partial\mathbf{E}}\mathcal{B}_{G\text{-}KL}(\mathbf{m}, \mathbf{C}, \mathbf{E}, c) = \sum_n \frac{\partial}{\partial s_n^2}\left\langle\log\phi(m_n + zs_n)\right\rangle\frac{\partial s_n^2}{\partial\mathbf{E}},$$

where the partial derivative with respect to s_n^2 is given in Section B.1.2 and

$$\frac{\partial s_n^2}{\partial\mathbf{E}} = \frac{\partial}{\partial\mathbf{E}}\mathbf{h}_n^\mathsf{T}\mathbf{E}\mathbf{C}^\mathsf{T}\mathbf{C}\mathbf{E}^\mathsf{T}\mathbf{h}_n = 2\mathbf{C}^\mathsf{T}\mathbf{C}\mathbf{E}^\mathsf{T}\mathbf{h}_n\mathbf{h}_n^\mathsf{T}.$$

B.2.3 Subspace optimisation : fixed point iteration

Another route to optimising the subspace vectors $\mathbf{E}$ is to use the form for optimal G-KL covariance matrix as presented in Section 4.3.1.1. Using this method, once we have optimised the bound with respect to $\{\mathbf{m}, \mathbf{C}_1, c\}$ we update the subspace vectors $\mathbf{E}$ to be the leading K eigen vectors of $\mathbf{S}$ as defined in equation (B.2.1). Whilst this procedure is not guaranteed to increase the bound in experiments it has yielded strong performance – see the results presented in Chapter 5.

For problems where the Gaussian potential has isotropic variance, $\mathbf{\Sigma} = \nu^2\mathbf{I}$, the form for the optimal G-KL inverse covariance, equation (4.3.1), simplifies to

$$\mathbf{S}^{-1} = \frac{1}{\nu^2}\mathbf{I} + \mathbf{H}\mathbf{\Gamma}\mathbf{H}^\mathsf{T}, \tag{B.2.1}$$

where $\mathbf{\Gamma}$ is defined in equation (4.3.2) of Section 4.3.2. We now consider two routes to updating the subspace vectors $\mathbf{E}$. First, we consider an approximate eigen decomposition method suitable for smaller non-sparse problems. Second, we consider an iterative Lanczos method better suited to larger sparse problems.

Approximate eigen decomposition. One route to possibly recovering the K leading eigenvectors of $\mathbf{S}$ is to evaluate the K smallest eigenvectors of $\frac{1}{\nu^2}\mathbf{I} + \mathbf{H}\mathbf{\Gamma}\mathbf{H}^\mathsf{T}$. We note that $\mathbf{H}\mathbf{\Gamma}\mathbf{H}^\mathsf{T} \approx \mathbf{H}\mathbf{\Gamma}'\mathbf{H}^\mathsf{T}$ where $\Gamma'_{nn} = \Gamma_{nn}$ if $\Gamma_{nn} > \delta$ and zero otherwise - we set δ small enough such that there are K non-zero diagonal elements $\mathbf{\Gamma}'$. If we now calculate the eigen decomposition to $\mathbf{H}\mathbf{\Gamma}'\mathbf{H}^\mathsf{T} = \mathbf{E}\mathbf{\Lambda}\mathbf{E}^\mathsf{T}$ we see that

$$\left[\frac{1}{\nu^2}\mathbf{I} + \mathbf{H}\mathbf{\Gamma}'\mathbf{H}^\mathsf{T}\right]^{-1} = \mathbf{E}\mathrm{diag}\left(\frac{\nu^2}{1+\lambda'_{nn}\nu^2}\right)\mathbf{E}^\mathsf{T}.$$

For $L \ll D$ we can evaluate the L eigenvectors of $\mathbf{H}\mathbf{\Gamma}'\mathbf{H}^\mathsf{T}$ cheaply since the eigenvalues of $\mathbf{X}\mathbf{X}^\mathsf{T}$ coincide with the eigenvalues of $\mathbf{X}^\mathsf{T}\mathbf{X}$[1]. Therefore approximating the K dimensional subspace eigen decomposition reduces to the complexity of decomposing a $K \times K$ matrix. If δ is small enough this method can often outperform approximate iterative decompositions provided the data is non-sparse and of moderate dimensionality.

Iterative Lanczos methods. Iterative Lanczos methods can approximately recover the eigen vectors corresponding to the largest and smallest eigen values of a matrix. General details about Lanczos methods can be found in Golub and Van Loan [1996], for the special case of covariance matrices of the form equation (B.2.1) details are provided in Seeger [2010]. Iterative Lanczos methods are fast provided the number of eigen vectors we wish to recover is not too large and matrix vector products can be computed efficiently – for example when the matrix has some special structure or is sparse.

B.3 Newton's method convergence rate conditions

Sufficient conditions under which optimising $\mathcal{B}_{G\text{-}KL}(\mathbf{m}, \mathbf{C})$ using Newtons method will exhibit quadratic convergence rates are that $\mathcal{B}_{G\text{-}KL}(\mathbf{m}, \mathbf{C})$ is twice continuously differentiable, strongly concave, has closed sublevel sets and has Lipschitz continuous Hessians on the sublevel sets [Boyd and Vandenberghe, 2004, section 9.5.3]. In Section 4.2.2 we showed that if all ϕ_n are log-concave then the

[1] To see this consider the eigen equation for $\mathbf{X}^\mathsf{T}\mathbf{X}\mathbf{E} = \mathbf{E}\mathbf{\Lambda}$ thus $\mathbf{X}\mathbf{X}^\mathsf{T}\mathbf{X}\mathbf{E} = \mathbf{X}\mathbf{E}\mathbf{\Lambda}$.

bound is strongly concave in $\mathbf{m}, \mathbf{C}$. In this section we provide conditions for which the other requirements hold. We consider G-KL inference problems of the form defined in Section 2.3 with $\{\phi_n\}_{n=1}^N$ site projection potentials that are piecewise exponentiated quadratics, log-concave and have unbounded support on $\mathbb{R}$. Specifically, we show that the required properties hold for potential functions that can be written

$$\phi(x) := \sum_{i=0}^{I} \mathbb{I}\left[x \in (l_i, l_{i+1})\right] \exp(a_i x^2 + b_i x + c_i),$$

where $-\infty = l_0 < l_1, ..., l_{I+1} = \infty$ and $\mathbb{I}\left[\cdot\right]$ is an indicator function equal to one when its argument is true and zero otherwise. Note that $\phi(x)$ need not be continuous and can have jump discontinuities at the partition points l_k. For such functions we have that $\log \phi(x) = \sum_{i=0}^{I} \mathbb{I}\left[x \in (l_i, l_{i+1})\right] a_i x^2 + b_i x + c_i$.

Continuously differentiable. The expectation of such potentials can then be expressed as a sum of integrals each over a disjoint domain

$$\left\langle \log \phi(\mathbf{w}^\mathsf{T}\mathbf{h}) \right\rangle = \sum_{i=0}^{I} \int_{l_i}^{l_{i+1}} \mathcal{N}\left(z|m, s^2\right) a_i z^2 + b_i z + c_i dz, \tag{B.3.1}$$

where $m = \mathbf{m}^\mathsf{T}\mathbf{h}$ and $s^2 = \|\mathbf{C}\mathbf{h}\|_2^2$. Each integral on the right hand side of equation (B.3.1) has a known analytic form which depends on terms of up to order 2 in m, s, standard normal density functions and cumulative density functions – see Marlin et al. [2011], Herbrich [2005] for their explicit forms and derivatives *w.r.t.* m, s. As an example, and to make this more concrete, we give the truncated expectation of just the quadratic term $a_i z^2$ below

$$\int_{l_i}^{l_{i+1}} a_i z^2 \mathcal{N}\left(z|m, s^2\right) dz$$
$$= a_i \left[s^2 \left(\tilde{l}_i \mathcal{N}(\tilde{l}_i) - \tilde{l}_{i+1} \mathcal{N}(\tilde{l}_{i+1}) \right) + \left(s^2 + m^2\right) \left(\Phi(\tilde{l}_{i+1}) - \Phi(\tilde{l}_i) \right) \right],$$

where $\tilde{l}_i := (l_i - m)/s$, $\mathcal{N}(x)$ is the standard normal density function and $\Phi(x)$ the standard normal cumulative distribution function. The truncated Gaussian expectation of the linear, $b_i z$, and the constant, c_i, terms have similar simpler analytic expressions.

We note that the standard normal density function and the standard normal cumulative density function are both smooth. Thus the expectation in equation (B.3.1) is the sum of smooth functions *w.r.t.* the parameters m, s. Therefore equation (B.3.1) as a function of $\mathbf{m}, \mathbf{C}$ is the composition of a function that is smooth in m, s and the functions $m = \mathbf{m}^\mathsf{T}\mathbf{h}$ and $s^2 = \|\mathbf{C}\mathbf{h}\|_2^2$ that are smooth in $\mathbf{m}, \mathbf{C}$. By the chain rule, we see that $\left\langle \log \phi(\mathbf{w}^\mathsf{T}\mathbf{h}) \right\rangle$ is smooth with respect to $\mathbf{m}, \mathbf{C}$.

By Lebesgues dominated convergence theorem, we expect the differentiability of $\left\langle \log \phi(\mathbf{w}^\mathsf{T}\mathbf{h}) \right\rangle$ to hold for a much broader class of potentials ϕ than the piecewise exponentiated quadratic class of functions considered here.

G-KL sublevel sets are closed. The G-KL sublevel sets, $\mathcal{S}$, are defined

$$\mathcal{S} := \left\{ \mathbf{m} \in \mathbb{R}^D, \mathbf{C} \in \mathbb{R}_{chol}^{D \times D} | \mathcal{B}(\mathbf{m}, \mathbf{C}) \geq \mathcal{B}(\mathbf{m}_0, \mathbf{C}_0) \right\}, \tag{B.3.2}$$

where $\mathbf{m}_0, \mathbf{C}_0$ are the moments that the G-KL bound optimisation procedure is initialised with and $\mathbb{R}_{Chol}^{D \times D}$ is the set of $D \times D$ upper triangular Cholesky matrices with strictly positive diagonals. Importantly $\mathcal{S}$ is closed since the G-KL bound is a closed function – which is a sufficient condition [Boyd and

Vandenberghe, 2004, p.471]. A function $f : \mathbb{R}^D \to \mathbb{R}$ with $\mathrm{dom}(f)$ open is closed iff f converges to $-\infty$ along every sequence converging to a boundary point of $\mathrm{dom}(f)$ [Boyd and Vandenberghe, 2004, p.640]. The G-KL bound is closed since it is the sum of the entropic term (which up to a constant is equal to $\sum_d \log(C_{dd})$), a negative quadratic in $\mathbf{m}, \mathbf{C}$, and $\langle \log \phi(\mathbf{w}^\mathsf{T}\mathbf{h}) \rangle$ (proven to be jointly concave in $\mathbf{m}, \mathbf{C}$). Thus for any sequence of moments $\{\mathbf{m}_k, \mathbf{C}_k\}$ that converges to the boundary of the G-KL domain we have $\mathcal{B}_{G\text{-}KL}(\mathbf{m}_k, \mathbf{C}_k)$ converging to $-\infty$.

G-KL Lipschitz continuous Hessians. We say the Hessian of f is Lipschitz continuous on $\mathcal{S}$ if there exists a constant $L \geq 0$ such that $\forall \mathbf{x}, \mathbf{y} \in \mathcal{S}$

$$\|\nabla^2 f(\mathbf{x}) - \nabla^2 f(\mathbf{y})\|_2 \leq L\|\mathbf{x} - \mathbf{y}\|_2. \tag{B.3.3}$$

An equivalent condition is that the Hessian has bounded and continuous derivatives on $\mathcal{S}$. Since the bound is continuously differentiable, since the sublevel sets are closed and since the entropys contribution to the bound ensures that s^2 is bounded below by a positive constant this property holds.

B.4 Complexity of bound and gradient computations

To perform G-KL approximate inference we optimise the G-KL bound, equation (4.1.1), by gradient ascent. In this section we consider the computational scaling properties of single evaluations of the bound and its gradient. We consider each term that depends on the variational parameters $\mathbf{m}$ and $\mathbf{S}$ separately, namely: $\log\det(\mathbf{S})$ from the entropy's contribution, $\mathrm{trace}\left(\boldsymbol{\Sigma}^{-1}\mathbf{S}\right)$ and $\mathbf{m}^\mathsf{T}\boldsymbol{\Sigma}^{-1}\mathbf{m}$ from the Gaussian potential's contribution, and $\left\{m_n, s_n^2\right\}_{n=1}^N$ from the product of site projection potential's contribution.

The G-KL covariance parameterisations we consider are: full Cholesky, diagonal Cholesky, banded Cholesky with bandwidth B, chevron Cholesky with K non-diagonal rows, subspace Cholesky with K dimensional subspace, sparse Cholesky with DK non-zeros, and factor analysis (FA) with K factor loading vectors. We report only the leading scaling terms and assume, for the sake of clarity, that $N \geq D \geq K, B$ where N is the number of site factors and D is the dimensionality of the parameter vector $\mathbf{w}$. In the last column we report the complexity figures required to compute the projected Gaussian moments $\{m_n, s_n^2\}_{n=1}^N$ where $m_n = \mathbf{m}^\mathsf{T}\mathbf{h}_n$, $s_n^2 = \|\mathbf{C}\mathbf{h}_n\|_2^2$, and $\mathtt{nnz} : \mathbb{R}^D \to \mathbb{N}$ is a function that counts the number of non-zero elements in a vector.

	$\log\det(\mathbf{S})$	$\mathrm{trace}\left(\boldsymbol{\Sigma}^{-1}\mathbf{S}\right)$			$\mathbf{m}^\mathsf{T}\boldsymbol{\Sigma}^{-1}\mathbf{m}$			$\{m_n, s_n^2\}_{n=1}^N$	
		$\boldsymbol{\Sigma}$ - iso	$\boldsymbol{\Sigma}$ - diag	$\boldsymbol{\Sigma}$ - full	$\boldsymbol{\Sigma}$ - iso	$\boldsymbol{\Sigma}$ - diag	$\boldsymbol{\Sigma}$ - full	$\mathtt{nnz}(\mathbf{h}) = D$	$\mathtt{nnz}(\mathbf{h}) = L$
$\mathbf{C}_{full}$	$O(D)$	$O(D^2)$	$O(D^2)$	$O(D^3)$	$O(D)$	$O(D)$	$O(D^2)$	$O(ND^2)$	$O(NDL)$
$\mathbf{C}_{diag}$	$O(D)$	$O(D)$	$O(D)$	$O(D)$	$O(D)$	$O(D)$	$O(D^2)$	$O(ND)$	$O(NL)$
$\mathbf{C}_{band}$	$O(D)$	$O(DB)$	$O(DB)$	$O(D^2B)$	$O(D)$	$O(D)$	$O(D^2)$	$O(NDB)$	$O(NLB)$
$\mathbf{C}_{chev}$	$O(D)$	$O(DK)$	$O(DK)$	$O(D^2K)$	$O(D)$	$O(D)$	$O(D^2)$	$O(NDK)$	$O(NLK)$
$\mathbf{C}_{sub}$	$O(K)$	$O(DK)$	$O(DK)$	$O(K^3)$	$O(D)$	$O(D)$	$O(D^2)$	$O(NK^2)$	$O(NK^2)$
$\mathbf{C}_{spar}$	$O(D)$	$O(DK)$	$O(DK)$	$O(D^2K)$	$O(D)$	$O(D)$	$O(D^2)$	$O(NDK)$	$O(NLK)$
$\mathbf{S}_{FA}$	$O(D^2K)$	$O(DK)$	$O(DK)$	$O(KD^2)$	$O(D)$	$O(D)$	$O(D^2)$	$O(NDK)$	$O(NLK)$

B.5 Transformation of Basis

When the model's Gaussian potential, $\mathcal{N}(\mathbf{w}|\boldsymbol{\mu}, \boldsymbol{\Sigma})$, has a full unstructured covariance matrix optimising the G-KL bound can sometimes be made less expensive by linearly transforming the basis of the param-

eters $\mathbf{m}$ and $\mathbf{C}$. To do this, essentially we hard code the information contributed to the posterior from the Gaussian potential into our G-KL parameters. That is we parameterise $\mathbf{m}$ and $\mathbf{C}$ using

$$\mathbf{C} = \tilde{\mathbf{C}}\mathbf{P} \quad \text{and} \quad \mathbf{m} = \mathbf{P}^\mathsf{T}\tilde{\mathbf{m}} + \boldsymbol{\mu}, \tag{B.5.1}$$

where $\mathbf{P}$ is the Cholesky decomposition of the prior covariance $\boldsymbol{\Sigma} = \mathbf{P}^\mathsf{T}\mathbf{P}$. Using this parameterisation each term of the G-KL bound simplifies such that:

$$\begin{aligned}
-\left\langle \log q(\mathbf{w}) \right\rangle &= \log\det\left(\tilde{\mathbf{C}}\right) + \log\det\left(\mathbf{P}\right) + \frac{D}{2}\log\left(2\pi\right) + \frac{D}{2}, \\
2\left\langle \log \mathcal{N}\left(\mathbf{w}|\boldsymbol{\mu}, \boldsymbol{\Sigma}\right) \right\rangle &= -D\log\left(2\pi\right) - D - 2\log\det\left(\mathbf{P}\right) - \tilde{\mathbf{m}}^\mathsf{T}\tilde{\mathbf{m}} - \operatorname{trace}\left(\tilde{\mathbf{C}}^\mathsf{T}\tilde{\mathbf{C}}\right), \\
\left\langle \psi(\mathbf{w}^\mathsf{T}\mathbf{h}) \right\rangle &= \int \mathcal{N}\left(z|0,1\right)\psi(m + zs)dz,
\end{aligned}$$

where $m := \tilde{\mathbf{m}}^\mathsf{T}\tilde{\mathbf{h}} + \boldsymbol{\mu}^\mathsf{T}\mathbf{h}$, $s := \|\tilde{\mathbf{C}}\tilde{\mathbf{h}}\|_2^2$ and $\tilde{\mathbf{h}} := \mathbf{P}\mathbf{h}$. Combining these terms the G-KL bound as a whole can be expressed as

$$\mathcal{B}(\mathbf{m}, \mathbf{C}) = \tilde{\mathcal{B}}(\tilde{\mathbf{m}}, \tilde{\mathbf{C}}) = \sum_d \log\left(\tilde{C}_{dd}\right) - \frac{1}{2}\tilde{\mathbf{m}}^\mathsf{T}\tilde{\mathbf{m}} - \frac{1}{2}\sum_{ij}\tilde{C}_{ij}^2 + \sum_n \left\langle \log\phi_n\left(m_n + zs_n\right)\right\rangle_{\mathcal{N}(z|0,1)}.$$

We are free then to optimise the transformed G-KL bound, $\tilde{\mathcal{B}}(\tilde{\mathbf{m}}, \tilde{\mathbf{C}})$, just with respect to $\tilde{\mathbf{m}}, \tilde{\mathbf{C}}$ at a reduced cost. For a model with a full covariance Gaussian potential and non-sparse $\mathbf{H} = [\mathbf{h}_1, ..., \mathbf{h}_N]$ computing the bound and gradient of $\tilde{\mathcal{B}}(\tilde{\mathbf{m}}, \tilde{\mathbf{C}})$ scales $O\left(D^2 + ND^2\right)$ whereas computing the bound and gradient of the untransformed bound scales $O\left(D^3 + ND^2\right)$ – see the table in Appendix B.4.

This procedure requires some pre-processing – namely the Cholesky decomposition of $\boldsymbol{\Sigma}$ and the 'whitening' of the dataset $\tilde{\mathbf{H}} = \mathbf{P}\mathbf{H}$ which scale $O\left(D^3\right)$ and $O\left(ND^2\right)$ respectively. And some post-processing – the final G-KL moments $\mathbf{m}$ and $\mathbf{C}$ are obtained using equations (B.5.1) which require a matrix-vector and a matrix-matrix product which scale $O\left(D^2\right)$ and $O\left(D^3\right)$ respectively.

Since during optimisation the bound and its gradient are usually computed many more times that twice, the basis transformation procedure detailed above will result in a significant computational saving. Note that this procedure can speed up G-KL bound optimisation only in settings where $\mathbf{h}_n$ are not sparse. For example Gaussian process regression models, where $\mathbf{h}_n$ are standard normal basis vectors, will not benefit from this reparameterisation since $\tilde{\mathbf{h}}_n = \mathbf{P}\mathbf{h}_n$ are not in general sparse.

B.6 Gaussian process regression

In this section we present some simple identities and approximations required to apply G-KL approximate inference to Gaussian process models. First we show how predictive densities can be approximated upon having approximated the posterior on the training data. Secondly we show how the G-KL bound's covariance hyperparameter derivatives can be computed for the Gaussian process model.

B.6.1 Predictive density

A Gaussian approximation to the posterior density on the latent function values of the training data may be used to obtain an approximation to the predictive density of the latent function value for a new test point. The GP predictive density to the target variable y_* for a new input $\mathbf{x}_*$ is defined by the integral

$$p(y_*|\mathbf{x}_*, \mathbf{X}, \mathbf{y}) = \int p(y_*|w_*)p(w_*|\mathbf{X}, \mathbf{y}, \mathbf{x}_*)dw_*. \tag{B.6.1}$$

The distribution on the test point latent function value, $p(w_*|\mathbf{X},\mathbf{y},\mathbf{x}_*)$, is approximated by marginalising out the training set latent variables using our Gaussian approximate posterior, $\mathcal{N}(\mathbf{w}|\mathbf{m},\mathbf{S}) \approx p(\mathbf{w}|\mathbf{X},\mathbf{y},\boldsymbol{\theta})$, giving

$$p(w_*|\mathbf{X},\mathbf{y},\mathbf{x}_*) = \int p(w_*|\mathbf{w},\mathbf{X},\mathbf{x}_*)p(\mathbf{w}|\mathbf{y},\mathbf{X})d\mathbf{w} \tag{B.6.2}$$

$$= \int \mathcal{N}\left(w_*|\boldsymbol{\sigma}_*^\mathsf{T}\boldsymbol{\Sigma}^{-1}\mathbf{w}, \sigma_{**} - \boldsymbol{\sigma}_*^\mathsf{T}\boldsymbol{\Sigma}^{-1}\boldsymbol{\sigma}_*\right) p(\mathbf{w}|\mathbf{y},\mathbf{X})d\mathbf{w} \tag{B.6.3}$$

$$\approx \int \mathcal{N}\left(w_*|\boldsymbol{\sigma}_*^\mathsf{T}\boldsymbol{\Sigma}^{-1}\mathbf{w}, \sigma_{**} - \boldsymbol{\sigma}_*^\mathsf{T}\boldsymbol{\Sigma}^{-1}\boldsymbol{\sigma}_*\right) \mathcal{N}(\mathbf{w}|\mathbf{m},\mathbf{S})\, d\mathbf{w} \tag{B.6.4}$$

$$= \mathcal{N}\left(w_*|\boldsymbol{\sigma}_*^\mathsf{T}\boldsymbol{\Sigma}^{-1}\mathbf{m}, \sigma_{**} - \boldsymbol{\sigma}_*^\mathsf{T}\boldsymbol{\Sigma}^{-1}\boldsymbol{\sigma}_* + \boldsymbol{\sigma}_*^\mathsf{T}\boldsymbol{\Sigma}^{-1}\mathbf{S}\boldsymbol{\Sigma}^{-1}\boldsymbol{\sigma}_*\right), \tag{B.6.5}$$

where $\boldsymbol{\sigma}_*$ and σ_{**} are the prior covariance and variance terms of the test data point $\mathbf{x}_*$. The elements of $\boldsymbol{\sigma}_*$ are calculated by evaluating the covariance function, $k(\mathbf{x},\mathbf{x}')$, between the each of the training covariates test point covariate such that $[\boldsymbol{\sigma}_*]_m = k(\mathbf{x}_m,\mathbf{x}_*)$ and $\sigma_{**} = k(\mathbf{x}_*,\mathbf{x}_*)$.

B.6.2 Hyperparameter optimisation

For a general likelihood $p(\mathbf{y}|\mathbf{w}) = \prod_{n=1}^N \phi_n(w_n)$ and GP prior $\mathcal{N}(\mathbf{w}|\mathbf{0},\boldsymbol{\Sigma})$ with covariance function $\Sigma_{mn} = k(\mathbf{x}_m,\mathbf{x}_n)$ we get the G-KL bound

$$\mathcal{B}_{G\text{-}KL}(\mathbf{m},\mathbf{C}) = \frac{D}{2} + \sum_n \log C_{nn} - \frac{1}{2}\log\det(\boldsymbol{\Sigma}) - \frac{1}{2}\mathbf{m}^\mathsf{T}\boldsymbol{\Sigma}^{-1}\mathbf{m} - \frac{1}{2}\mathrm{trace}\left(\boldsymbol{\Sigma}^{-1}\mathbf{S}\right) + \sum_n \left\langle \log\phi(m_n + z\sqrt{S_{nn}})\right\rangle_{\mathcal{N}(z|0,1)}. \tag{B.6.6}$$

Taking the derivative of the above expression with respect to the covariance hyperparameters $\boldsymbol{\theta}$ we get

$$\frac{\partial\mathcal{B}_{G\text{-}KL}}{\partial\boldsymbol{\theta}} = -\frac{1}{2}\mathrm{trace}\left(\boldsymbol{\Sigma}^{-1}\frac{\partial\boldsymbol{\Sigma}}{\partial\boldsymbol{\theta}}\right) + \frac{1}{2}\mathbf{m}^\mathsf{T}\boldsymbol{\Sigma}^{-1}\frac{\partial\boldsymbol{\Sigma}}{\partial\boldsymbol{\theta}}\boldsymbol{\Sigma}^{-1}\mathbf{m} + \frac{1}{2}\mathrm{trace}\left(\mathbf{C}\boldsymbol{\Sigma}^{-1}\frac{\partial\boldsymbol{\Sigma}}{\partial\boldsymbol{\theta}}\boldsymbol{\Sigma}^{-1}\mathbf{C}\right). \tag{B.6.7}$$

Note that $\mathbf{m}$ and $\mathbf{C}$ implicitly depend on the covariance hyperparameters $\boldsymbol{\theta}$. However, cross terms such as

$$\frac{\partial\mathcal{B}_{G\text{-}KL}}{\partial\mathbf{m}}\frac{\partial\mathbf{m}}{\partial\boldsymbol{\theta}} \quad \text{or} \quad \frac{\partial\mathcal{B}_{G\text{-}KL}}{\partial\mathbf{C}}\frac{\partial\mathbf{C}}{\partial\boldsymbol{\theta}} \tag{B.6.8}$$

do not contribute to equation (B.6.7) at the optimum of the G-KL bound since the gradients of $\mathcal{B}_{G\text{-}KL}$ with respect to $\mathbf{m}$ or $\mathbf{C}$ are zero at this point. Therefore, to evaluate the gradient of $\mathcal{B}_{G\text{-}KL}$ with respect to the covariance hyperparameters first the G-KL bound is optimised with respect to $\mathbf{m},\mathbf{C}$ with $\boldsymbol{\theta}$ fixed, then at that optimum we use equation (B.6.7) to calculate the derivative with respect to $\boldsymbol{\theta}$.

B.7 Original concavity derivation

The concavity proof presented in Section 4.2.2 is due to Michalis Titsias, which provides a cleaner presentation of the original result we made in Challis and Barber [2011]. For completeness, below we present our original derivation.

For log-concave potentials $\phi(x)$ we show that the G-KL bound $\mathcal{B}_{G\text{-}KL}(\mathbf{m},\mathbf{S})$, equation (4.1.1), is jointly concave with respect to the variational Gaussian parameters $\mathbf{m}$ and $\mathbf{C}$ where $\mathbf{C}$ is the Cholesky decomposition of the covariance such that $\mathbf{S} = \mathbf{C}^\mathsf{T}\mathbf{C}$.

Since the bound depends on the logarithm of ϕ, without loss of generality we may take $N = 1$, and on ignoring constants with respect to $\mathbf{m}$ and $\mathbf{S}$, we have that

$$\mathcal{B}_{G\text{-}KL}(\mathbf{m}, \mathbf{C}) \overset{c.}{=} \sum_i \log C_{ii} - \frac{1}{2}\mathbf{m}^\mathsf{T}\boldsymbol{\Sigma}^{-1}\mathbf{m} + \boldsymbol{\mu}^\mathsf{T}\boldsymbol{\Sigma}^{-1}\mathbf{m} - \frac{1}{2}\mathrm{trace}\left(\boldsymbol{\Sigma}^{-1}\mathbf{C}^\mathsf{T}\mathbf{C}\right) + \left\langle \log \phi(\mathbf{w}^\mathsf{T}\mathbf{h}) \right\rangle \quad \text{(B.7.1)}$$

Excluding $\left\langle \log \phi(\mathbf{w}^\mathsf{T}\mathbf{h}) \right\rangle$ from the expression above, all terms are concave functions exclusively in either $\mathbf{m}$ or $\mathbf{C}$. Since the sum of concave functions on distinct variables is jointly concave, these terms represent a jointly concave contribution. To complete the proof we therefore need to show that $\left\langle \log \phi(\mathbf{w}^\mathsf{T}\mathbf{h}) \right\rangle$ is jointly concave in $\mathbf{m}$ and $\mathbf{C}$. We first transform variables to write $\left\langle \log \phi(\mathbf{w}^\mathsf{T}\mathbf{h}) \right\rangle$ as

$$\left\langle \log \phi(a) \right\rangle_{\mathcal{N}(a|\mathbf{m}^\mathsf{T}\mathbf{h}, \mathbf{h}^\mathsf{T}\mathbf{S}\mathbf{h})} = \left\langle \psi(\mu(\mathbf{m}) + z\sigma(\mathbf{C})) \right\rangle_z \quad \text{(B.7.2)}$$

where $\langle \cdot \rangle_z$ refers to taking the expectation with respect to the standard normal $\mathcal{N}(z|0,1)$ and, $\mu(\mathbf{m}) := \mathbf{m}^\mathsf{T}\mathbf{h}$, $\sigma(\mathbf{C}) := \sqrt{\mathbf{h}^\mathsf{T}\mathbf{C}^\mathsf{T}\mathbf{C}\mathbf{h}}$ and $\psi := \log \phi$. Note that establishing the concavity of equation (B.7.2) is non-trivial since the function $\psi(\mu(\mathbf{m}) + z\sigma(\mathbf{C}))$ is itself *not* jointly concave in $\mathbf{C}$ and $\mathbf{m}$.

For ease of notation we let $\sigma' := \mathrm{vec}\left(\frac{\partial \sigma(\mathbf{C})}{\partial \mathbf{C}}\right)$, where $\mathrm{vec}(\mathbf{X})$ is the vector obtained by concatenating the columns of $\mathbf{X}$, with dimension D^2; $\sigma'' := \frac{\partial^2 \sigma(\mathbf{C})}{\partial \mathbf{C}}$ is the Hessian of σ with respect to $\mathbf{C}$ with dimension $D^2 \times D^2$; $\mu' := \frac{\partial \mu(\mathbf{m})}{\partial \mathbf{m}}$ is a column vector with dimension D. Then the Hessian of ψ with respect to $\mathbf{m}$ and $\mathbf{C}$ can be expressed in the following block matrix form

$$H[\psi] = \begin{bmatrix} \frac{\partial^2 \psi}{\partial \mathbf{C}} & \frac{\partial^2 \psi}{\partial \mathbf{C} \partial \mathbf{m}} \\ \frac{\partial^2 \psi}{\partial \mathbf{m} \partial \mathbf{C}} & \frac{\partial^2 \psi}{\partial \mathbf{m}} \end{bmatrix} = \begin{bmatrix} \psi'' z^2 \sigma' \sigma'^\mathsf{T} + \psi' z \sigma'' & \psi'' z \sigma' \mu'^\mathsf{T} \\ \psi'' z \mu' \sigma'^\mathsf{T} & \psi'' \mu' \mu'^\mathsf{T} \end{bmatrix}$$

The Hessian of $\left\langle \psi(\mu(\mathbf{m}) + z\sigma(\mathbf{C})) \right\rangle_z$ is equivalent to $\left\langle H[\psi(\mu(\mathbf{m}) + z\sigma(\mathbf{C}))] \right\rangle_z$, which we now show to be negative semi-definite. Since the expectation in $\left\langle H[\psi(\mu(\mathbf{m}) + z\sigma(\mathbf{C}))] \right\rangle_z$ is with respect to an even Gaussian density function, provided that for all $\gamma \geq 0$, the combined Hessian defined as

$$H_{z=-\gamma}[\psi] + H_{z=+\gamma}[\psi] \preceq 0 \quad \text{(B.7.3)}$$

is negative definite then the expectation of $H[\psi]$ with respect to z is negative definite. To show this we first note that for all $\mathbf{u} \in \mathbb{R}^{D^2}$ and $\mathbf{v} \in \mathbb{R}^D$

$$\begin{bmatrix} \mathbf{u} \\ \mathbf{v} \end{bmatrix}^\mathsf{T} H[\psi] \begin{bmatrix} \mathbf{u} \\ \mathbf{v} \end{bmatrix} = \psi'' \left[\mathbf{v}^\mathsf{T}\mu' + z\mathbf{u}^\mathsf{T}\sigma'\right]^2 + \psi' z \mathbf{u}^\mathsf{T}\sigma''\mathbf{u}$$

The first term of the right hand side is negative for all values of z since $\psi''(x) \leq 0$. To show that equation (B.7.3) is satisfied it is sufficient to show that

$$\left(\psi'(\mu + \gamma\sigma) - \psi'(\mu - \gamma\sigma)\right) \gamma \mathbf{u}^\mathsf{T}\sigma''\mathbf{u} \leq 0$$

which is true since $\sigma'' \succeq 0$, $\sigma(\mathbf{C}) \geq 0$ and because $\psi'(x)$ is a decreasing function from the assumed log-concavity of ϕ.

To see that $\sigma'' \succeq 0$ we write, $\sigma^2(\mathbf{C}) = \sum_j g_j^2(\mathbf{C})$ where $g_j(\mathbf{C}) = |\sum_i h_i C_{ij}|$ is convex and non-negative for all j. For convex and non-negative functions g_j and $p > 1$, then $\left(\sum_{j=1}^W g_j(x)^p\right)^{1/p}$ is convex [Boyd and Vandenberghe, 2004], which reveals that $\sigma(\mathbf{C})$ is convex on setting $p = 2$.

B.8 vgai documentation

A Matlab implementation of the G-KL approximate inference methods described in this thesis is publicly available via the mloss.org website at mloss.org/software/view/308/. The vgai package implements G-KL approximate inference for latent linear models of the form described in Section 2.3. The toolbox includes implementations of a selection of non-Gaussian site-projection potentials – see Table B.1. Generic non-Gaussian site-projection potentials are supported if an implementation of $\psi : \mathbb{R} \to \mathbb{R}$ where $\psi := \log \phi$ is provided. The package implements the constrained concave parameterisations of covariance discussed in Section 4.3.1.3 and the factor analysis parameterisation. G-KL bound optimisation is achieved in the vgai implementation using Mark Schmidt's minFunc optimisation package.[2]

B.8.1 Code structure

The vgai package has two core data structures that define the G-KL approximate inference problem: the vg Matlab struct which specifies the properties of the Gaussian approximation to the target density, and the pots Matlab cell of structs which specifies the intractable target density. Below we review how each of these data structures must be defined and used in the vgai Matlab package.

Specifying the Gaussian approximation : vg

The vg structure is used to store and specify the variational Gaussian approximation, $q(\mathbf{w}) = \mathcal{N}(\mathbf{w}|\mathbf{m}, \mathbf{S})$, to the target density $p(\mathbf{w})$. The Gaussian variational density $q(\mathbf{w}) = \mathcal{N}(\mathbf{w}|\mathbf{m}, \mathbf{S})$, with $\mathbf{w} \in \mathbb{R}^D$ the parameters of interest and $\mathbf{m} \in \mathbb{R}^D$, $\mathbf{S} \in \mathbb{R}^{D \times D}$ the Gaussian's mean and covariance, is stored and specified as a Matlab struct variable with the following fields:

vg.dim Is an integer D that specifies the dimension of the parameter vector $\mathbf{w}$ (and thus also the dimensionality of $\mathbf{m}, \mathbf{S}$). There is no default value for this field – the user must set its value before optimising the G-KL bound.

vg.m Is a $D \times 1$ column vector specifying the G-KL mean $\mathbf{m}$. Unless specified its default initialisation is the zero vector.

vg.param Is a string that specifies which parameterisation of G-KL covariance is used. Its default setting is 'full' corresponding to the full Cholesky parameterisation. The constrained parameterisations can be specified using: 'band' for a banded Cholesky, 'chev' for a chevron Cholesky, 'sub' for subspace Cholesky and 'fa' for a factor analysis parameterised covariance. The additional fields associated with each of these parameterisations are described below:

- **full** $\mathbf{S} = \mathbf{C}^{\mathsf{T}}\mathbf{C}$ with $\mathbf{C}$ an upper-triangular Cholesky matrix. $\mathbf{C}$ is stored and accessed as vg.c. The default initialisation for $\mathbf{C}$ is the identity matrix.
- **band** $\mathbf{S} = \mathbf{C}_{Band}^{\mathsf{T}}\mathbf{C}_{Band}$ where $\mathbf{C}_{Band}$ is an upper-triangular banded Cholesky matrix with bandwidth K. The banded diagonal elements are stored in a tall matrix such that $[\mathbf{C}_{band}]_{i,j} = [\mathbf{B}]_{j,j-i+1}$ if $0 < j - i < K$ otherwise $[\mathbf{C}_{band}]_{i,j} = 0$. Banded Cholesky

[2] The minFunc optimisation package can be downloaded from www.di.ens.fr/~mschmidt/Software/minFunc.html.

covariances require the bandwidth K to be specified in the field `vg.bw`. The matrix $\mathbf{B}$ is specified using the field `vg.b`. The default initialisation corresponds to setting $\mathbf{C}_{band}$ to the identity matrix.

chev $\mathbf{S} = \mathbf{C}_{chev}^{\mathsf{T}}\mathbf{C}_{chev}$. A chevron Cholesky matrix with K non-diagonal rows is defined such that $[\mathbf{C}_{chev}]_{ij} = [\mathbf{\Theta}]_{i,j}$ if $j \leq i \leq K$, or $[\mathbf{C}_{chev}]_{ij} = d_i$ if $i = j$ or zero otherwise. The number of non-diagonal rows K needs to be specified in the field `vg.k`. $\mathbf{\Theta}^{\mathsf{T}}$ is stored in the field `vg.t`. The default initialisation corresponds to setting $\mathbf{C}_{chev}$ to the identity matrix.

sub $\mathbf{S} = \mathbf{E}_1\mathbf{C}^{\mathsf{T}}\mathbf{C}\mathbf{E}_1^{\mathsf{T}} + c^2\mathbf{E}_2\mathbf{E}_2^{\mathsf{T}}$, where $\mathbf{E}_1 \in \mathbb{R}^{D\times K}$ is the orthonormal subspace basis vectors such that $\mathbf{E}_1^{\mathsf{T}}\mathbf{E}_1 = \mathbf{I}_{K\times K}$, $\mathbf{C} \in \mathbb{R}^{K\times K}$ is the subspace Cholesky matrix, c^2 is the off-subspace isotropic variance and $\mathbf{E}_2$ refers to the off-subspace basis vectors that do not need to be computed or maintained (as explained in Appendix B.2). The subspace parameterisation of covariance requires the specification of K the 'rank' of the parameterisation in the field `vg.k`. The field `vg.cs` stores the $K \times K$ subspace Cholesky matrix $\mathbf{C}$, the field `vg.ci` stores the off-subspace isotropic standard deviation c, and `vg.es` stores the $D \times K$ orthonormal subspace basis vectors $\mathbf{E}_1$. The default initialisation is $\mathbf{C} = \mathbf{I}_{K\times K}$, $c = 1$ and $\mathbf{E}_1$ is constructed as a vertical concatenation of K-dimensional identity matrices.

fa $\mathbf{S} = \mathbf{\Theta}\mathbf{\Theta}^{\mathsf{T}} + \mathrm{diag}\left(\mathbf{d}^2\right)$ where $\mathbf{\Theta} \in \mathbb{R}^{D\times K}$ and $\mathbf{d} \in \mathbb{R}^{D\times 1}$. The factor analysis parameterisation of covariance requires the 'rank' K of the parameterisation to be specified in the field `vg.k`. The matrix $\mathbf{\Theta}$ is stored in the field `vg.t` and the column vector $\mathbf{d}$ is stored in the field `vg.d`. The default initialisation is $\mathbf{d} = \mathbf{1}$ and $\mathbf{\Theta}$ constructed as a vertical concatenation of $\mathbf{I}_{K\times K}$ matrices.

Specifying the inference problem : `pots`

We consider solving inference problems where the target density $p(\mathbf{w})$ can be defined by the product

$$p(\mathbf{w}) = \frac{1}{Z}\prod_{m=1}^{M}\boldsymbol{\phi}_m(\mathbf{w}), \tag{B.8.1}$$

where each factor $\boldsymbol{\phi}_m(\mathbf{w})$ is itself a product of site-projection potentials or is a multivariate Gaussian potential such that:

$$\boldsymbol{\phi}_m(\mathbf{w}) := \prod_{n=1}^{N_m}\phi_n(\mathbf{w}^{\mathsf{T}}\mathbf{h}_n^m), \quad \text{or} \quad \boldsymbol{\phi}_m(\mathbf{w}) := \mathcal{N}\left(\mathbf{H}^{m\mathsf{T}}\mathbf{w}|\boldsymbol{\mu}, \mathbf{\Sigma}\right).$$

In this section we refer each factor $\boldsymbol{\phi}_m(\mathbf{w})$ as a group potential. To enforce consistency between the Gaussian and the non-Gaussian group potentials we define $\mathbf{H}^m := [\mathbf{h}_1^m, \ldots, \mathbf{h}_{N_m}^m] \in \mathbb{R}^{D\times N_m}$. The Gaussian group potential defined above has mean $\boldsymbol{\mu} \in \mathbb{R}^{N_m}$ and covariance $\mathbf{\Sigma} \in \mathbb{R}^{N_m\times N_m}$.

To specify an inference problem of the form of equation (B.8.1) in the `vgai` package we use the `pots` variable which is a `cell` of `structs`. The m^{th} element of the `pots cell` is a `pot struct` which defines the m^{th} group potential $\boldsymbol{\phi}_m(\mathbf{w})$. In the Bayesian generalised linear models considered in this thesis typically $M = 2$, where $\boldsymbol{\phi}_1(\mathbf{w})$ describes the collection of potentials that define the prior and the second group potential $\boldsymbol{\phi}_2(\mathbf{w})$ describes the collection of potentials that define the likelihood. Below we show how either a Gaussian or a non-Gaussian site-projection group potential can be defined.

Group potential : product of site-projection potentials. For $\boldsymbol{\phi}_m(\mathbf{w}) := \prod_{n=1}^{N_m} \phi(\mathbf{w}^\mathsf{T}\mathbf{h}_n^m)$ we define the m^{th} element of the `pots` cell using `pots{m}=pot`, where `pot` is a Matlab `struct` with the following fields:

`pot.type` User specified string that has to be set to the value `'prodPhi'`.

`pot.dim` User specified two element row vector such that `pot.dim(1)` $= D$ the dimensionality of $\mathbf{w}$ and `pot.dim(2)` $= N_m$ the number of site-projection potentials.

`pot.logphi` User specified function handle to the function that evaluates $\log \phi(x)$. For example for a logistic regression likelihood this should be set to `pot.logphi=@log_siglogit`.

`pot.params` Optional structure of parameters that is passed to the function that evaluates $\log \phi(x) : \mathbb{R} \to \mathbb{R}$. Default value is null.

`pot.logphi_c` Optional user specified function that evaluates normalisation constants of potential functions that are constant when taking the log potential's expectation with respect to $\mathbf{w}$. This function must take only the `pot.params struct` as its argument. The returned value is added to each evaluation of $\log \phi$. The default value is the zero function.

`pot.numint` Optional user specified structure that defines the numerical integration procedure used to evaluate the site-projection potential expectations.

Group potential : multivariate Gaussian potential. For $\boldsymbol{\phi}_m(\mathbf{w}) := \mathcal{N}\left(\mathbf{H}^{m\mathsf{T}}\mathbf{w}|\boldsymbol{\mu}, \boldsymbol{\Sigma}\right)$ a multivariate Gaussian potential we define the m^{th} element of the `pots` cell using `pots{m}=pot`, where `pot` is defined using the fields:

`pot.type` User specified string that has to be set to the value `'gaussian'`.

`pot.dim` User specified two element row vector such that `pot.dim(1)` $= D$ the dimensionality of $\mathbf{w}$ and `pot.dim(2)` $= N_m$ such that $\mathbf{H}^m \in \mathbb{R}^{D \times N_m}$.

`pot.H` This field defines the $\mathbf{H}^m \in \mathbb{R}^{D \times N_m}$ matrix. If $D = N_m$ then this field is optional with its default value the D-dimensional identity matrix. If $D \neq N_m$ this field must be specified by the user.

`pot.mu` Vector specifying the $N_m \times 1$ Gaussian mean vector $\boldsymbol{\mu}$. Default value is the zero vector.

`pot.cov` Optional specification of the Gaussian $N_m \times N_m$ covariance matrix. This field can be specified by a scalar which corresponds to a scaling of the identity matrix, an $N_m \times 1$ column vector which corresponds to a diagonal covariance, or a full $N_m \times N_m$ covariance matrix.

Demo code : Bayesian logistic regression

The following short Matlab script generates some synthetic data, defines the G-KL inference problem and performs G-KL approximate inference using the `vgai` package.

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% PROBLEM PARAMETERS
D     = 100;  % data dimension
Ntrn  = 200;  % no. of training instances
Ntst  = 500;  % no. of test instances
nu    = 0.2;  % fraction of miss labeled data

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% GENERATE SYNTHETIC DATA
wtr   = randn(D,1);               % true data generating weight vector

X = randn(D,Ntrn+Ntst);                   % covariates X(d,n) ~ N(0,1)
Y = sign(X'*wtr);                        % class labels y_n \in {-1,1}
flipy = rand(Ntrn+Ntst,1)<nu;  Y(flipy)=-Y(flipy);   % add label noise

Xtrn=X(:,1:Ntrn); Xtst=X(:,Ntrn+1:end);          % training test split
Ytrn=Y(1:Ntrn);   Ytst=Y(Ntrn+1:end);

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% DEFINE THE INFERENCE PROBLEM
% Potential 1 is the Gaussian distributed prior on the weights:
%     pot{1} := \prod_d N(w_d|0,1).
%
% vgai assumes a default Gaussian mean = 0 and identity covariance
% matrix.
pot{1}.type    = 'gaussian';
pot{1}.dim     = [D D];
%
% Potential 2 is the likelihood:
%   pot{2} := \prod_n^N p(y_n|x_n,w),
% where p(y_n|w'*x_n)=sig(y_n*w'*x_n) and sig(x) is the logistic
% sigmoid function sig(x)=1./(1+exp(-x)).
pot{2}.type    = 'prodPhi';
pot{2}.logphi  = @log_siglogit;        % :=log phi(x). where x=w'*h_n
pot{2}.dim     = [D Ntrn];
pot{2}.H       = bsxfun(@times,Xtrn,Ytrn');% H = [h_1,...,h_(n-1),h_n]

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% PERFORM G-KL APPROXIMATE INFERENCE
% Define the vg structure.
vgo.dim=D;                 % specify dimension of variational Gaussian

[vg logZBoundTrace] = vgopt(pot,vgo);         % optimise the G-KL bound
```

Demo code : sparse latent linear model

The following short Matlab script generates some synthetic data from a sparse latent linear model and then performs G-KL approximate inference.

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% PROBLEM PARAMETERS
D     = 50;           % data dimension
D0    = 20;           % no. of zeros in weight vector used to sample data
Ntrn  = D;            % no. training instances
Ntst  = 500;          % no. of test instances
```

Potential function	Matlab name	Evaluation method	Parameters	Section ref.
Logistic$(v\|m,s)$	logistic	numeric	m,s	Section A.5.1
Laplace$(v\|m,s)$	laplace	analytic	m,s	Section A.5.2
Student$(v\|\nu,m,s)$	stut	numeric	m,s,nu	Section A.5.3
Cauchy$(v\|m,s)$	cauchy	numeric	m,s	Section A.5.4
$\sigma_{logit}(m)$	siglogit	numeric		Section A.5.5
$\sigma_{probit}(m)$	sigprobit	numeric		Section A.5.6
$\sigma_{heavi}(m)$	sigheavi	analytic	eps	Section A.5.7

Table B.1: A list of potential functions implemented in the vgai package.

```
s2    = 0.01;         % observation noise variance
tau   = 1./sqrt(2);   % Laplace prior variance

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% GENERATE SYNTHETIC DATA
wtr   = randn(D,1);             % sample data generating weight vector
wtr(1:D0)=0; wtr=wtr(randperm(D));
X     = randn(D,Ntrn+Ntst); % sample observation matrix X(d,n) ~ N(0,1)
Y     = X'*wtr + sqrt(s2).*randn(Ntrn+Ntst,1);   % y_n ~ N(w'*x_n|0,s2)

Xtrn=X(:,1:Ntrn); Xtst=X(:,Ntrn+1:end);           % training test split
Ytrn=Y(1:Ntrn);   Ytst=Y(Ntrn+1:end);

%%%%%%%%%%%%%%%%%%%%%%%%% SPARSE LINEAR MODEL WITH A STUDENT'S T PRIOR
% Potential 1 is the sparse Laplace prior on the weight vectors w:
%   pot{1} := \prod_d phi_d(w_d),
% where phi_d := 1/(2*tau_d) e^(-|w_d|/tau_d)
pot{1}.type      = 'prodPhi';   % define the potential type.
pot{1}.int       = 'analytic';  % integral is performed analytically.
pot{1}.explogphi = @exp_log_laplace; % :=<log phi(mu_n+z*sigma_n)>_N(w|m,S)
pot{1}.dim       = [D D];
pot{1}.params.tau= tau; % gives unit variance.

% Potential 2 is the Gaussian likelihood:
% pot{2} := N(y|H'*w,s^2)
pot{2}.type       = 'gaussian';
pot{2}.H          = Xtrn;
pot{2}.mu         = Ytrn;
pot{2}.dim        = [D Ntrn];
pot{2}.cov        = s2; % isotropic gaussian covariance

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% PERFORM G-KL APPROXIMATE INFERENCE
vg_full.dim = D;        % Need to specify dim. of variational Gaussian.

[vg logZTrace]=vgopt(pot,vg_full);              % optimise the gkl bound.
```

Appendix C

Affine independent KL approximate inference

C.1 AI-KL bound and gradients

In this section we describe how to efficiently numerically evaluate the AI-KL bound and associated gradients with respect to the parameters $\mathbf{A} = \mathbf{LU}, \mathbf{b}$ and $\boldsymbol{\theta}$.

C.1.1 Entropy

The entropy's contribution to the AI-KL bound can be written as

$$H\left[q_{\mathbf{w}}(\mathbf{w}|\mathbf{A},\mathbf{b},\boldsymbol{\theta})\right] = \log\left|\det\left(\mathbf{A}\right)\right| + \sum_{d=1}^{D} H\left[q_{v_d}(v_d|\theta_d)\right], \tag{C.1.1}$$

where $H\left[q(v_d|\theta_d)\right]$ is the univariate differential entropy of the base density $q_{v_d}(v_d|\theta_d)$. The partial derivatives of equation (C.1.1) with respect to $\mathbf{A}$ are given by

$$\frac{\partial}{\partial \mathbf{A}} H\left[q_{\mathbf{w}}(\mathbf{w}|\mathbf{A},\mathbf{b},\boldsymbol{\theta})\right] = \mathbf{A}^{-\mathsf{T}}.$$

The derivatives of each of the marginal base density's entropies, $H\left[q(v_d|\theta_d)\right]$, depend on the parametric form of the chosen base density $q_{v_d}(v_d|\theta_d)$ and the parameter θ_d only. For the results presented in Chapter 6, only two base densities were used: the skew-normal and the generalised-normal. The entropy and respective derivatives of these base distributions are presented in Section A.5.

For the LU parameterised bound, such that $\mathbf{A} = \mathbf{LU}$ with $\mathbf{L}$ lower-triangular and $\mathbf{U}$ upper-triangular matrices, we have

$$\log\left|\det\left(\mathbf{A}\right)\right| = \sum_{d=1}^{D} \log L_{dd} + \log U_{dd}.$$

Thus the partial derivatives of the entropy with respect to $\mathbf{L}$ and $\mathbf{U}$ are given by

$$\frac{\partial}{\partial L_{mn}} H\left[q_{\mathbf{w}}(\mathbf{w}|\mathbf{A},\mathbf{b},\boldsymbol{\theta})\right] = \delta_{mn}\frac{1}{L_{mn}}, \quad \text{and} \quad \frac{\partial}{\partial U_{mn}} H\left[q_{\mathbf{w}}(\mathbf{w}|\mathbf{A},\mathbf{b},\boldsymbol{\theta})\right] = \delta_{mn}\frac{1}{U_{mn}},$$

where δ_{mn} is the Kronecker delta.

C.1.2 Site projection potentials

In the main text we showed that the expectation of $\psi\left(\mathbf{h}^{\mathsf{T}}\mathbf{w}\right)$ with respect to $q_{\mathbf{w}}(\mathbf{w}|\mathbf{A},\mathbf{b},\boldsymbol{\theta})$ can be efficiently computed by using the FFT. In this section first we review this result making clear each step of the derivation. Second, we show how the derivatives of $\left\langle\psi\left(\mathbf{h}^{\mathsf{T}}\mathbf{w}\right)\right\rangle$ with respect to $\mathbf{A},\mathbf{b},\boldsymbol{\theta}$ can also be efficiently computed.

Computing the expectation

The expectation $\left\langle \psi(\mathbf{h}^\mathsf{T}\mathbf{w}) \right\rangle_{q_\mathbf{w}(\mathbf{w})}$ for $\psi : \mathbb{R} \to \mathbb{R}$ some non-linear non-quadratic function, $\mathbf{h} \in \mathbb{R}^D$ some fixed vector and $q_\mathbf{w}(\mathbf{w}|\mathbf{A}, \mathbf{b}, \boldsymbol{\theta})$ an AI density is equivalent to the univariate expectation $\langle \psi(y) \rangle_{q_y(y)}$ where the density $q_y(y)$ can be expressed as the convolution of the scaled base densities $\{q_{u_d}(u_d|\theta_d)\}_{d=1}^D$. To show this, first we write the expectation of $\psi(\mathbf{h}^\mathsf{T}\mathbf{w})$ with respect to $q_\mathbf{w}(\mathbf{w}|\mathbf{A}, \mathbf{b}, \boldsymbol{\theta})$ as an expectation with respect to $q_\mathbf{v}(\mathbf{v})$,

$$\left\langle \psi\left(\mathbf{h}^\mathsf{T}\mathbf{w}\right) \right\rangle_{q_\mathbf{w}(\mathbf{w})} = \int \psi\left(\mathbf{h}^\mathsf{T}\mathbf{w}\right) q_\mathbf{w}(\mathbf{w}) d\mathbf{w} = \int \psi\left(\mathbf{h}^\mathsf{T}\mathbf{A}\mathbf{v} + \mathbf{h}^\mathsf{T}\mathbf{b}\right) q_\mathbf{v}(\mathbf{v}) d\mathbf{v}, \tag{C.1.2}$$

above, and in what follows, to simplify notation we omit the conditional terms $\mathbf{A}, \mathbf{b}, \boldsymbol{\theta}$ from the AI density $q_\mathbf{w}(\mathbf{w}|\mathbf{A}, \mathbf{b}, \boldsymbol{\theta})$. The last equality in equation (C.1.2) is obtained by making the substitution $\mathbf{w} = \mathbf{A}\mathbf{v} + \mathbf{b}$. Again, for a cleaner notation, in what follows we let $\boldsymbol{\alpha} = \mathbf{A}^\mathsf{T}\mathbf{h}$ and $\beta = \mathbf{h}^\mathsf{T}\mathbf{b}$. We now substitute $\psi(\boldsymbol{\alpha}^\mathsf{T}\mathbf{v} + \beta) = \int \delta(y - \boldsymbol{\alpha}^\mathsf{T}\mathbf{v} - \beta)\psi(y)dy$, where $\delta(x)$ is the Dirac delta function, into equation (C.1.2) to give us

$$\left\langle \psi\left(\mathbf{w}^\mathsf{T}\mathbf{h}\right) \right\rangle_{q_\mathbf{w}(\mathbf{w})} = \int \psi(y) \int \prod_d q_{v_d}(v_d|\theta_d) \delta\left(y - \boldsymbol{\alpha}^\mathsf{T}\mathbf{v} - \beta\right) d\mathbf{v} dy = \langle \psi(y) \rangle_{q_y(y)}. \tag{C.1.3}$$

In equation (C.1.3) above $q_y(y)$ is the density of the random variable y defined as the linear projection of the random variables $\mathbf{v}$ such that $y = \boldsymbol{\alpha}^\mathsf{T}\mathbf{v} + \beta$. Thus the univariate marginal density $q_y(y)$ is defined by the integral

$$q_y(y) := \int \delta\left(y - \boldsymbol{\alpha}^\mathsf{T}\mathbf{v} - \beta\right) \prod_d q_{v_d}(v_d|\theta_d) d\mathbf{v}.$$

Whilst this integral is generally intractable we can make the substitution $\delta(x) = \int e^{2\pi i t x} dt$ to give us

$$q_y(y) = \int \int e^{2\pi i t\left(y - \boldsymbol{\alpha}^\mathsf{T}\mathbf{v} - \beta\right)} \prod_d q_{v_d}(v_d|\theta_d) d\mathbf{v} dt \tag{C.1.4}$$

$$= \int e^{2\pi i t y} e^{-2\pi i t \beta} \prod_d \int e^{-2\pi i t \alpha_d v_d} q_{v_d}(v_d|\theta_d) dv_d dt. \tag{C.1.5}$$

We now inspect each term in equation (C.1.5). First we consider an individual factor of the group product:

$$\begin{aligned} \int q_{v_d}(v_d) e^{-2\pi i t \alpha_d v_d} dv_d &= \frac{1}{|\alpha_d|} \int q_{v_d}\left(\frac{u_d}{\alpha_d}\right) e^{-2\pi i t u_d} du_d \\ &= \int q_{u_d}\left(u_d|\theta_d\right) e^{-2\pi i t u_d} du_d =: \tilde{q}_{u_d}(t), \end{aligned}$$

where the first equality above comes from making the substitution $u_d = \alpha_d v_d$. This substitution defines the univariate density $q_{u_d}(u_d|\theta_d) = \frac{1}{|\alpha_d|} q_{v_d}(\frac{u_d}{\alpha_d}|\theta_d)$. Thus each factor of the group product in equation (C.1.5) is the Fourier transform of the density $q_{u_d}(u_d|\theta_d)$. The $e^{-2\pi i t \beta}$ factor corresponds to the Fourier transform of a delta mean shift $\delta(y - \beta)$. Putting this together, equation (C.1.5) can be interpreted as the inverse Fourier transform of the product of the Fourier transforms of $\{q_{u_d}(u_d|\theta_d)\}_{d=1}^D$ and of the β mean shift. Algebraically this gives us an expression for the marginal $q_y(y)$ in the form

$$q_y(y) = \int e^{2\pi i t y} e^{-2\pi i t \beta} \prod_d \tilde{q}_{u_d}(t) dt. \tag{C.1.6}$$

This result is a reworking of the D-fold convolution theorem for probability densities. We provide the derivation here so that it may form the basis of subsequent derivations required to evaluate the AI-KL bound's derivatives as univariate integrals.

Numerical evaluation

Since only in very special cases we have simple analytic forms for the univariate density $q_y(y)$ we resort to numerical methods to evaluate it. To do so we evaluate equation (C.1.6) replacing $\{q_{u_d}(u_d|\theta_d)\}_{d=1}^D$ with their discrete lattice approximations $\{\hat{q}_{u_d}(u_d|\theta_d)\}_{d=1}^D$. We now show that making this substitution results in $\hat{q}_y(y)$ as defined in equation(2.6) which can be efficiently computed by utilising the FFT algorithm.

First, we must define the set of lattice points used to evaluate the discrete approximate densities $\{\hat{q}_{u_d}(u_d|\theta_d)\}_{d=1}^D$. The user defines the number of lattice points $K \in \mathbb{N}$ according to their computational budget or accuracy requirements. The accuracy can be roughly assessed by computing the difference in the expectation using K and $2K$ lattice points. The lattice end points are chosen such that $[l_1, l_K] = [-\nu\sigma_y, \nu\sigma_y]$ where σ_y is the standard deviation of the random variable y given by $\sigma_y^2 = \sum_d \alpha_d^2 \mathrm{var}(v_d)$. ν is a user defined parameter, in our experiments we set $\nu = 6$ and double K until the bound value changes by less than 10^{-3}. The lattice points $[l_1, ..., l_K]$ are evenly spaced such that $\Delta = l_{k+1} - l_k$ is constant for all k.

The continuous Fourier transform of the lattice density $\hat{q}_{u_d}(u_d|\theta_d)$ takes the form

$$\tilde{\hat{q}}_{u_d}(t) := \int e^{-2\pi i t u_d} \hat{q}_{u_d}(u_d|\theta_d) du_d = \sum_{k=1}^{K} \pi_{dk} e^{-2\pi i t l_k}.$$

Taking the inverse Fourier transform of the product of these transforms, as $q(y)$ is defined in equation (C.1.6), we get

$$\begin{aligned}
\hat{q}_y(y) &= \int e^{2\pi i t(y-\beta)} \prod_d \sum_{k_d=1}^{K} \pi_{dk_d} e^{-2\pi i t l_{k_d}} dt \\
&= \sum_{[k_1,\ldots,k_D]} \int e^{2\pi i t\left(y-\beta-\sum_d l_{k_d}\right)} \prod_d \pi_{dk_d} dt \\
&= \sum_{[k_1,\ldots,k_D]} \delta\left(y - \beta - \sum_d l_{k_d}\right) \prod_d \pi_{dk_d}, \qquad \text{(C.1.7)}
\end{aligned}$$

where the sum over $[k_1, ..., k_D]$ in equation (C.1.7) refers to the sum over the K^D permutations of the D dimensional cartesian product of lattice point indices $[k_1, ..., k_D]$. We note that $l_{k_d} = l_k$, the subscript is only to distinguish the different permutations of the sum.

Equation(C.1.7) describes a mixture of delta distributions and is the exact result from computing the convolution of the lattice approximate densities by means of the continuous Fourier transform. Importantly, the K^D mixtures in equation (C.1.7) collapse to just DK distinct delta points since l_k are evenly spaced.

When $D = 2$:

$$\hat{q}_y(y) = \sum_{j=1}^{K} \sum_{k=1}^{K} \pi_{1j} \pi_{2k} \delta(y - \beta - l_j - l_k). \qquad \text{(C.1.8)}$$

We can see from equation (C.1.8) above that $\hat{q}_y(y)$ is a mixture of $2K$ delta densities evenly spaced at lattice points $[2l_1, ..., 2l_K]$,

$$\hat{q}_y(y) = \sum_{n=1}^{2K} \rho_n \delta(y - \beta - l_n)$$

for suitably defined $\boldsymbol{\rho}$. For a single lattice point l_m,

$$\rho_m = \sum_{i,j:i+j=m} \pi_{1j}\pi_{2k} = \sum_{n=1}^{2K} \pi'_{1n}\pi'_{2(m-n)} = \left[\texttt{ifft}\left[\texttt{fft}[\boldsymbol{\pi}'_1] \cdot \texttt{fft}[\boldsymbol{\pi}'_2]\right]\right]_m ,$$

Here $\boldsymbol{\pi}'$ refers to the zero padded vector of delta mixture weights $\boldsymbol{\pi}' = [\boldsymbol{\pi}, \mathbf{0}]$ such that $\mathbf{0}$ is a K dimensional vector of zeros. If $m - n < 1$ we extend the indices $\boldsymbol{\pi}'_{m-n} := \boldsymbol{\pi}'_{2K+m-n}$; this extension is valid and does not affect the convolution due to the zero padding of $\boldsymbol{\pi}'$. The last equality in the expression above is the statement of the discrete Fourier transform convolution theorem.

The result can be extended to higher dimensions $D > 2$ by induction, using the associativity of the convolution operator and the fact that lattice point locations are invariant to convolution, to give

$$\hat{q}_y(y) = \sum_{n=1}^{DK} \rho_n \delta(y - \beta - l_n) \quad \text{where} \quad \boldsymbol{\rho} = \texttt{ifft}\left[\prod_d \texttt{fft}\left[\boldsymbol{\pi}'_d\right]\right],$$

For general D, $\boldsymbol{\pi}'$ refers to the zero padded vector of delta mixture weights $\boldsymbol{\pi}' = [\boldsymbol{\pi}, \mathbf{0}]$ such that $\mathbf{0}$ is a $(D-1)K$ dimensional vector of zeros.

Partial derivatives : A

Taking the partial derivative of $\left\langle \psi\left(\mathbf{w}^\mathsf{T}\mathbf{h}\right)\right\rangle$ with respect to A_{mn} we obtain

$$\frac{\partial}{\partial A_{mn}} \left\langle \psi(\mathbf{w}^\mathsf{T}\mathbf{h})\right\rangle = h_n \int q_{\mathbf{v}}(\mathbf{v})\psi'\left(\mathbf{h}^\mathsf{T}\mathbf{A}\mathbf{v} + \mathbf{b}^\mathsf{T}\mathbf{h}\right) v_m d\mathbf{v}.$$

As previously mentioned the above form is not equivalent to $h_n \left\langle v_m \psi'(y)\right\rangle_{q_y(y)}$. It can, however, still be expressed as a one dimensional integral:

$$\begin{aligned}
\frac{\partial}{\partial A_{mn}} \left\langle \psi\left(\mathbf{w}^\mathsf{T}\mathbf{h}\right)\right\rangle &= h_n \int v_m \prod_{d=1}^{D} q_{v_d}(v_d|\theta_d)\psi'\left(\mathbf{h}^\mathsf{T}\mathbf{A}\mathbf{v} + \mathbf{b}^\mathsf{T}\mathbf{h}\right) d\mathbf{v} \\
&= h_n \int v_m \prod_{d=1}^{D} q_{v_d}(v_d|\theta_d) \int \delta\left(y - \boldsymbol{\alpha}^\mathsf{T}\mathbf{v} - \beta\right)\psi'(y) dy d\mathbf{v} \\
&= h_n \int v_m q_{v_m}(v_m|\theta_m) \prod_{d\neq m} q_{v_d}(v_d|\theta_d) \int \delta\left(y - \boldsymbol{\alpha}^\mathsf{T}\mathbf{v} - \beta\right)\psi'(y) dy d\mathbf{v} \\
&= h_n \int \psi'(y) \int v_m q_{v_m}(v_m|\theta_m) \prod_{d\neq m} q_{v_d}(v_d|\theta_d)\delta\left(y - \boldsymbol{\alpha}^\mathsf{T}\mathbf{v} - \beta\right) d\mathbf{v} dy
\end{aligned}$$

where $\psi'(y) := \frac{d}{dy}\psi(y)$, $\boldsymbol{\alpha} := \mathbf{A}^\mathsf{T}\mathbf{h}$ and $\beta := \mathbf{b}^\mathsf{T}\mathbf{h}$ as before. To evaluate the expression above we define the univariate weighting function $d_m(y)$, such that

$$d_m(y) := \int v_m q_{v_m}(v_m|\theta_m) \prod_{d\neq m} q_{v_d}(v_d|\theta_d)\delta\left(y - \boldsymbol{\alpha}^\mathsf{T}\mathbf{v} - \beta\right) d\mathbf{v}.$$

Using this weighting function the gradient can be expressed simply as

$$\frac{\partial}{\partial A_{mn}} \left\langle \psi\left(\mathbf{w}^\mathsf{T}\mathbf{h}\right)\right\rangle = h_n \int \psi'(y) d_m(y) dy.$$

We evaluate $d_m(y)$ by means of computing its Fourier transform. The Fourier transform of $d_m(y)$ is given by

$$\begin{aligned}\tilde{d}_m(t) &= \int e^{-2\pi i t y} \int v_m q_{v_m}(v_m|\theta_m) \prod_{d\neq m} q_{v_d}(v_d|\theta_d)\delta\left(y - \boldsymbol{\alpha}^\mathsf{T}\mathbf{v} - \beta\right) d\mathbf{v}dy \\ &= e^{-2\pi i t\beta} \int v_m q_{v_m}(v_m|\theta_m) e^{-2\pi i t\alpha_m v_m} \prod_{d\neq m} q_{v_d}(v_d|\theta_d) e^{-2\pi i t\alpha_d v_d} d\mathbf{v} \\ &= e^{-2\pi i t\beta} \times \tilde{e}_m(t) \times \prod_{d\neq m} \tilde{q}_{u_d}(t|\theta_d)\end{aligned}$$

where $\tilde{e}_m(t|\theta_m)$ is the Fourier transform of the univariate expectation

$$\tilde{e}_m(t|\theta_m) := \int v_m q_{v_m}(v_m|\theta_m) e^{-2\pi i t\alpha_m v_m} dv_m = \int \frac{u_m}{\alpha_m} q_{u_m}(u_m|\theta_m)\, e^{-2\pi i t u_m} du_m.$$

Partial derivatives : $\mathbf{b}$

Taking the partial derivative of $\left\langle \psi\left(\mathbf{w}^\mathsf{T}\mathbf{h}\right)\right\rangle$ with respect to b_m we get

$$\begin{aligned}\frac{\partial}{\partial b_m}\left\langle \psi\left(\mathbf{w}^\mathsf{T}\mathbf{h}\right)\right\rangle &= h_m \int \prod_{d=1}^{D} q_{v_d}(v_d|\theta_d)\psi'\left(\mathbf{h}^\mathsf{T}\mathbf{A}\mathbf{v} + \mathbf{b}^\mathsf{T}\mathbf{h}\right) d\mathbf{v} \\ &= h_m \int q_y(y)\psi'(y)dy.\end{aligned}$$

Partial derivatives : $\boldsymbol{\theta}$

Taking the derivative of $\left\langle \psi\left(\mathbf{w}^\mathsf{T}\mathbf{h}\right)\right\rangle$ with respect to θ_m we get

$$\begin{aligned}\frac{\partial}{\partial \theta_m}\left\langle \psi\left(\mathbf{w}^\mathsf{T}\mathbf{h}\right)\right\rangle &= \frac{\partial}{\partial \theta_m}\int \prod_{d=1}^{D} q_{v_d}(v_d|\theta_d)\psi\left(\mathbf{h}^\mathsf{T}\mathbf{A}\mathbf{v} + \mathbf{b}^\mathsf{T}\mathbf{h}\right) d\mathbf{v} \\ &= \int \left[\frac{\partial}{\partial \theta_m} q_{v_m}(v_m|\theta_m)\right] \prod_{d\neq m} q_{v_d}(v_d|\theta_d)\psi\left(\mathbf{h}^\mathsf{T}\mathbf{A}\mathbf{v} + \mathbf{b}^\mathsf{T}\mathbf{h}\right) d\mathbf{v} \\ &= \int \psi(y) \int \left[\frac{\partial}{\partial \theta_m} q_{v_m}(v_m|\theta_m)\right] \prod_{d\neq m} q_{v_d}(v_d|\theta_d)\delta\left(y - \mathbf{h}^\mathsf{T}\mathbf{A}\mathbf{v} - \mathbf{b}^\mathsf{T}\mathbf{h}\right) dy d\mathbf{v}\end{aligned}$$

Similar to the gradient of $\left\langle \psi_n(\mathbf{w}^\mathsf{T}\mathbf{h}_n)\right\rangle$ with respect to A_{mn} we define a derivative weighting function $\tilde{p}'_d$ such that

$$\begin{aligned}\tilde{p}'_d(t) &:= \int e^{-2\pi i t y} \int \left[\frac{\partial}{\partial \theta_m} q_{v_m}(v_m|\theta_m)\right] \prod_{d\neq m} q_{v_d}(v_d|\theta_d)\delta\left(y - \mathbf{h}^\mathsf{T}\mathbf{A}\mathbf{v} - \mathbf{b}^\mathsf{T}\mathbf{h}\right) dy d\mathbf{v} \\ &= e^{-2\pi i t\beta}\left[\prod_{d\neq m} \tilde{q}_{u_d}(t|\theta_d)\right] \int e^{-2\pi i t\alpha_m v_m} \frac{\partial}{\partial \theta_m} p(v_m|\theta_m) dv_m.\end{aligned}$$

For $p'_d(y)$ the inverse Fourier transform of $\tilde{p}'_d(t)$ we obtain the gradient

$$\frac{\partial}{\partial \theta_m}\left\langle \psi\left(\mathbf{w}^\mathsf{T}\mathbf{h}\right)\right\rangle = \int p'_d(y)\psi(y)dy.$$

C.1.3 Gaussian potentials

For the Gaussian potential $\mathcal{N}(\mathbf{w}|\boldsymbol{\mu}, \boldsymbol{\Sigma})$, its log expectation under $q_\mathbf{w}(\mathbf{w}|\mathbf{A}, \mathbf{b}, \boldsymbol{\theta})$ is given by

$$2\left\langle \log \mathcal{N}(\mathbf{w}|\boldsymbol{\mu}, \boldsymbol{\Sigma})\right\rangle = -D\log 2\pi - \log\det(\boldsymbol{\Sigma}) - \left\langle \mathbf{w}^\mathsf{T}\boldsymbol{\Sigma}^{-1}\mathbf{w}\right\rangle + 2\left\langle \mathbf{w}\right\rangle \boldsymbol{\Sigma}^{-1}\boldsymbol{\mu} - \boldsymbol{\mu}^\mathsf{T}\boldsymbol{\Sigma}^{-1}\boldsymbol{\mu}.$$

To evaluate this expression we precompute the Cholesky decomposition of Gaussian precision matrix $\boldsymbol{\Sigma}^{-1} = \mathbf{P}^\mathsf{T}\mathbf{P}$, which scales $O\left(D^3\right)$ and only needs to be performed once. Since $\langle \mathbf{w} \rangle = \mathbf{A}\langle \mathbf{v} \rangle + \mathbf{b}$ and $\left\langle \mathbf{v}^\mathsf{T}\mathbf{B}\mathbf{v} \right\rangle = \langle \mathbf{v} \rangle^\mathsf{T} \mathbf{B} \langle \mathbf{v} \rangle + \mathrm{trace}\left(\mathbf{B}\mathrm{cov}(\mathbf{v})\right)$ we have that

$$\begin{aligned}\left\langle \mathbf{w}^\mathsf{T}\boldsymbol{\Sigma}^{-1}\mathbf{w} \right\rangle &= \left\langle \mathbf{v}^\mathsf{T}\mathbf{A}^\mathsf{T}\boldsymbol{\Sigma}^{-1}\mathbf{A}\mathbf{v} \right\rangle + 2\left\langle \mathbf{v}^\mathsf{T}\mathbf{A}^\mathsf{T}\boldsymbol{\Sigma}^{-1}\mathbf{b} \right\rangle + \mathbf{b}^\mathsf{T}\boldsymbol{\Sigma}^{-1}\mathbf{b}, \\ &= \langle \mathbf{v} \rangle^\mathsf{T} \mathbf{A}^\mathsf{T}\boldsymbol{\Sigma}^{-1}\mathbf{A}\langle \mathbf{v} \rangle + \mathrm{trace}\left(\mathbf{A}^\mathsf{T}\boldsymbol{\Sigma}^{-1}\mathbf{A}\mathrm{cov}(\mathbf{v})\right) \\ &\qquad + 2\langle \mathbf{v} \rangle^\mathsf{T} \mathbf{A}^\mathsf{T}\boldsymbol{\Sigma}^{-1}\mathbf{b} + \mathbf{b}^\mathsf{T}\boldsymbol{\Sigma}^{-1}\mathbf{b} \\ \langle \mathbf{w} \rangle \boldsymbol{\Sigma}^{-1}\boldsymbol{\mu} &= \langle \mathbf{v} \rangle^\mathsf{T} \mathbf{A}^\mathsf{T}\boldsymbol{\Sigma}^{-1}\boldsymbol{\mu} + \mathbf{b}^\mathsf{T}\boldsymbol{\Sigma}^{-1}\boldsymbol{\mu}.\end{aligned}$$

Since $\mathbf{v}$ are assumed independent we define $\mathbf{D} := \mathrm{cov}(\mathbf{v}) = \mathrm{diag}\left(\mathrm{var}(\mathbf{v})\right)$. All terms in the expression above, except for the trace term, can be computed as a sequence of matrix vector products. To compute the trace term we use $\mathrm{trace}\left(\mathbf{A}^\mathsf{T}\boldsymbol{\Sigma}^{-1}\mathbf{A}\mathrm{cov}(\mathbf{v})\right) = \mathrm{vec}\left(\mathbf{PLUD}^{\frac{1}{2}}\right)^\mathsf{T} \mathrm{vec}\left(\mathbf{PLUD}^{\frac{1}{2}}\right)$, where $\mathrm{vec}\left(\mathbf{X}\right)$ constructs a column vector by concatenating the columns of the matrix $\mathbf{X}$ and $\mathbf{D}^{\frac{1}{2}}$ is the square root of the diagonal covariance matrix, which scale $O\left(D^3\right)$ for general $\boldsymbol{\Sigma}$. When $\boldsymbol{\Sigma} = \sigma^2\mathbf{I}$ this reduces to $O\left(D^2\right)$.

Partial derivatives : $\mathbf{A}$

The partial derivatives of the Gaussian potential's contribution to the AI-KL bound, as detailed above, with respect to $\mathbf{A}$ and $\mathbf{b}$ are

$$\begin{aligned}\frac{\partial}{\partial \mathbf{A}} 2\left\langle \log \mathcal{N}\left(\mathbf{w}|\boldsymbol{\mu}, \boldsymbol{\Sigma}\right) \right\rangle = \frac{\partial}{\partial \mathbf{A}} &- \langle \mathbf{v} \rangle^\mathsf{T} \mathbf{A}^\mathsf{T}\boldsymbol{\Sigma}^{-1}\mathbf{A}\langle \mathbf{v} \rangle - \mathrm{trace}\left(\mathbf{A}^\mathsf{T}\boldsymbol{\Sigma}^{-1}\mathbf{A}\mathbf{D}\right) \\ &- 2\langle \mathbf{v} \rangle^\mathsf{T} \mathbf{A}^\mathsf{T}\boldsymbol{\Sigma}^{-1}\mathbf{b} + 2\langle \mathbf{v} \rangle^\mathsf{T} \mathbf{A}^\mathsf{T}\boldsymbol{\Sigma}^{-1}\boldsymbol{\mu} + 2\mathbf{b}^\mathsf{T}\boldsymbol{\Sigma}^{-1}\boldsymbol{\mu},\end{aligned}$$

which can be expressed as

$$\frac{\partial}{\partial \mathbf{A}} \left\langle \log \mathcal{N}\left(\mathbf{w}|\boldsymbol{\mu}, \boldsymbol{\Sigma}\right) \right\rangle = -\boldsymbol{\Sigma}^{-1}\mathbf{A}\left(\langle \mathbf{v} \rangle \langle \mathbf{v} \rangle^\mathsf{T} + \mathbf{D}\right) + \langle \mathbf{v} \rangle \left(\boldsymbol{\mu}\boldsymbol{\Sigma}^{-1} - \boldsymbol{\Sigma}^{-1}\mathbf{b}\right)^\mathsf{T},$$

and can be computed using sequential matrix vector products and vector outer products.

Partial derivatives : $\mathbf{b}$

The partial derivative of the Gaussian potential's contribution to the AI-KL bound with respect to $\mathbf{b}$ is given by

$$\begin{aligned}\frac{\partial}{\partial \mathbf{b}} 2\left\langle \log \mathcal{N}\left(\mathbf{w}|\boldsymbol{\mu}, \boldsymbol{\Sigma}\right) \right\rangle &= \frac{\partial}{\partial \mathbf{b}} - 2\langle \mathbf{v} \rangle^\mathsf{T} \mathbf{A}^\mathsf{T}\boldsymbol{\Sigma}^{-1}\mathbf{b} - \mathbf{b}^\mathsf{T}\boldsymbol{\Sigma}^{-1}\mathbf{b} + 2\mathbf{b}^\mathsf{T}\boldsymbol{\Sigma}^{-1}\boldsymbol{\mu} \\ &= -2\boldsymbol{\Sigma}^{-1}\left(\mathbf{A}\langle \mathbf{v} \rangle + \mathbf{b} + \boldsymbol{\mu}\right).\end{aligned}$$

Partial derivatives : $\mathbf{L}, \mathbf{U}$

We extend the above results to the LU decomposition of the transformation matrix such that $\mathbf{A} = \mathbf{LU}$ where $\mathbf{L}$ is lower-triangular and $\mathbf{U}$ upper-triangular matrices. To do so, we apply the chain rule, noting that $A_{mn} = \sum_k L_{mk}U_{kn}$, to give us

$$\frac{\partial A_{mn}}{\partial L_{uv}} = \delta_{mu}U_{vn} \quad \text{and} \quad \frac{\partial A_{mn}}{\partial U_{st}} = \delta_{tn}L_{ms}$$

for δ_{ij} the Kronecker delta. Thus to compute the derivative of $F(\mathbf{A}) = F(\mathbf{LU})$ we have that

$$\frac{\partial}{\partial L_{uv}} F(\mathbf{A}) = \sum_{mn} \frac{\partial}{\partial A_{mn}} F(\mathbf{A}) \delta_{mu} U_{vn} \quad \text{when} \quad u \geq v \quad \text{and zero otherwise}$$
$$\frac{\partial}{\partial U_{st}} F(\mathbf{A}) = \sum_{mn} \frac{\partial}{\partial A_{mn}} F(\mathbf{A}) \delta_{tn} L_{ms} \quad \text{when} \quad t \geq s \quad \text{and zero otherwise.}$$

C.2 Blockwise concavity

Here we present a simple reworking, and extension, of the concavity result provided in Chapter 4 for log-concave potentials $\{\phi_n\}_{n=1}^N$ and Gaussian KL approximate inference. Whilst the AI-KL bound is jointly concave in $\mathbf{L}$ and $\mathbf{b}$ or $\mathbf{U}$ and $\mathbf{b}$ it is not jointly concave in $\mathbf{L}$ and $\mathbf{U}$ simultaneously.

The entropy of the AI bound is clearly concave in both $\mathbf{L}$ and $\mathbf{U}$ being a sum of log terms acting on individual elements of $\mathbf{L}$ and $\mathbf{U}$.

The Gaussian potential's contribution to the AI bound is a negative quadratic in $\mathbf{L}$ or $\mathbf{U}$. To see this we consider the Gaussian contribution, omitting constants *w.r.t.* $\mathbf{U}$, $\mathbf{L}$ and $\mathbf{b}$ we have that

$$2\left\langle \log \mathcal{N}\left(\mathbf{w}|\boldsymbol{\mu}, \boldsymbol{\Sigma}\right)\right\rangle \overset{c.}{=} -\bar{\mathbf{v}}^\mathsf{T}\mathbf{U}^\mathsf{T}\mathbf{L}^\mathsf{T}\boldsymbol{\Sigma}^{-1}\mathbf{L}\mathbf{U}\bar{\mathbf{v}} - \operatorname{trace}\left(\mathbf{U}^\mathsf{T}\mathbf{L}^\mathsf{T}\boldsymbol{\Sigma}^{-1}\mathbf{L}\mathbf{U}\mathbf{D}\right) - 2\bar{\mathbf{v}}^\mathsf{T}\mathbf{U}^\mathsf{T}\mathbf{L}^\mathsf{T}\boldsymbol{\Sigma}^{-1}\mathbf{b} - \mathbf{b}^\mathsf{T}\boldsymbol{\Sigma}^{-1}\mathbf{b} + 2\bar{\mathbf{v}}\mathbf{U}^\mathsf{T}\mathbf{L}^\mathsf{T}\boldsymbol{\Sigma}^{-1}\boldsymbol{\mu} + 2\mathbf{b}^\mathsf{T}\boldsymbol{\Sigma}^{-1}\boldsymbol{\mu}$$

where $\bar{\mathbf{v}} = \langle \mathbf{v} \rangle$ and $\mathbf{D} = \operatorname{diag}\left(\operatorname{var}(\mathbf{v})\right)$. Keeping $\mathbf{L}$ fixed and denoting $\mathbf{X} = \mathbf{L}^\mathsf{T}\boldsymbol{\Sigma}^{-1}\mathbf{L}$ we get

$$2\left\langle \log \mathcal{N}\left(\mathbf{w}|\boldsymbol{\mu}, \boldsymbol{\Sigma}\right)\right\rangle \overset{c.}{=} -\bar{\mathbf{v}}^\mathsf{T}\mathbf{U}^\mathsf{T}\mathbf{X}\mathbf{U}\bar{\mathbf{v}} - \operatorname{trace}\left(\mathbf{U}^\mathsf{T}\mathbf{X}\mathbf{U}\mathbf{D}\right) - 2\bar{\mathbf{v}}^\mathsf{T}\mathbf{U}^\mathsf{T}\mathbf{L}^\mathsf{T}\boldsymbol{\Sigma}^{-1}\mathbf{b} - \mathbf{b}^\mathsf{T}\boldsymbol{\Sigma}^{-1}\mathbf{b} + 2\bar{\mathbf{v}}\mathbf{U}^\mathsf{T}\mathbf{L}^\mathsf{T}\boldsymbol{\Sigma}^{-1}\boldsymbol{\mu} + 2\mathbf{b}^\mathsf{T}\boldsymbol{\Sigma}^{-1}\boldsymbol{\mu}$$

which is a negative quadratic in $\mathbf{U}$ and $\mathbf{b}$ and is thus jointly concave in these parameters. A similar analysis carries through for $\mathbf{L}$ keeping $\mathbf{U}$ fixed.

Without loss of generality we can consider the concavity of a single non-linear site potential's contribution to the AI-KL bound. For a single site potential we define

$$\mathcal{E}(\mathbf{A}, \mathbf{b}) := \langle \log \phi_n(\mathbf{w}) \rangle = \int q_{\mathbf{v}}(\mathbf{v}) \psi(\mathbf{h}^\mathsf{T}\mathbf{A}\mathbf{v} + \mathbf{b}^\mathsf{T}\mathbf{h}) d\mathbf{v}$$

where $\psi(x) = \log \phi(x)$ for a non-Gaussian site potential $\phi(x) : \mathbb{R} \to \mathbb{R}^+$, and $\phi(x)$ is assumed log-concave so that $\forall \theta \in [0, 1]$ we have

$$\psi\left(\theta x + (1-\theta)y\right) \geq \theta\psi(x) + (1-\theta)\psi(y).$$

Thus considering two affine parameter settings $\{\mathbf{A}_1, \mathbf{b}_1\}$ and $\{\mathbf{A}_2, \mathbf{b}_2\}$ we have that

$$\mathcal{E}\left(\theta\mathbf{A}_1 + (1-\theta)\mathbf{A}_2, \theta\mathbf{b}_1 + (1-\theta)\mathbf{b}_2\right) = \left\langle \psi\left(\theta\left(\mathbf{h}^\mathsf{T}\mathbf{A}_1\mathbf{v} + \mathbf{b}_1^\mathsf{T}\mathbf{h}\right) + (1-\theta)\left(\mathbf{h}^\mathsf{T}\mathbf{A}_2\mathbf{v} + \mathbf{b}_2^\mathsf{T}\mathbf{h}\right)\right)\right\rangle,$$

from the concavity of ψ, and the linearity of the expectation operator, we have

$$\mathcal{E}(\theta\mathbf{A}_1 + (1-\theta)\mathbf{A}_2, \theta\mathbf{b}_1 + (1-\theta)\mathbf{b}_2) \geq \theta\left\langle \psi(\mathbf{h}^\mathsf{T}\mathbf{A}_1\mathbf{v} + \mathbf{b}_1^\mathsf{T}\mathbf{h})\right\rangle + (1-\theta)\left\langle \psi(\mathbf{h}^\mathsf{T}\mathbf{A}_2\mathbf{v} + \mathbf{b}_2^\mathsf{T}\mathbf{h})\right\rangle,$$

and so the non-Gaussian site potentials contribute terms that are concave in $\mathbf{A}$ to the AI-KL bound. Concavity in $\mathbf{L}$ follows through by letting $\mathbf{h} = \mathbf{U}\mathbf{h}$, similarly the converse holds for concavity in $\mathbf{U}$ keeping $\mathbf{L}$ fixed.

C.3 AI base densities

In this section we specify the base densities used to construct affine independent multivariate densities for the experiments presented in Chapter 6. Specifically we present the generalised-normal and the skew-normal base densities, providing equations to compute the density's entropy, partial derivatives and moments.

C.3.1 Skew-normal

For a skew-normal distributed random variable, $v \sim \mathcal{SN}(m, s, \nu)$, we parameterise its density using

$$\mathcal{SN}(v|m, s, \nu) = \frac{2}{s}\mathcal{N}(r)\Phi(\nu r), \quad \text{where} \quad r := \frac{v-m}{s},$$

$\mathcal{N}(z) = \mathcal{N}(z|0, 1)$, $\Phi(z) = \int_{-\infty}^{z} \phi(x)dx$, with location parameter $m \in \mathbb{R}$, scale parameter $s \in \mathbb{R}^+$ and skew parameter $\nu \in \mathbb{R}$. When $\nu = 0$ the skew-normal density is equivalent to the univariate Gaussian density $\mathcal{N}\left(v|m, s^2\right)$.

Moments. The moments of the skew-normal density are: $\langle v \rangle = m + s\delta\sqrt{2/\pi}$, $\mathrm{var}(v) = s^2\left(1 - \frac{2\delta^2}{\pi}\right)$ where $\delta = \frac{\nu}{\sqrt{1+\nu^2}}$, with skew and excess kurtosis defined

$$\mathrm{skw}(v) = \frac{4-\pi}{2}\frac{\left(d\sqrt{2/\pi}\right)^3}{\left(1-2\delta^2/\pi\right)^{3/2}}, \quad \mathrm{kur}(v) = 2(\pi-3)\frac{\left(\delta\sqrt{2/\pi}\right)^4}{\left(1-2\delta^2/\pi\right)^2}.$$

Derivatives. The derivative of the log of the skew-normal density with respect to ν is given by

$$\frac{\partial}{\partial\nu}\log\mathcal{SN}(v|m, s, \nu) = \frac{r\phi(\nu r)}{\Phi(r)} \quad \text{where} \quad r = \frac{v-m}{s}. \tag{C.3.1}$$

Similarly to the generalised-normal base density, we do not need to consider optimising the AI-KL bound with respect to the scale and location parameters of the skew-normal since for $q_{\mathbf{w}}(\mathbf{w})$ location and scale is already parameterised by $\mathbf{A}, \mathbf{b}$.

Entropy. We are not aware of an analytic form for the skew-normal density's entropy. Therefore we used univariate rectangular quadrature to compute these terms.

C.3.2 Generalised-normal

For a generalised-normal distributed random variable, $v \sim \mathcal{GN}(m, s, \eta)$, we parameterise its density using

$$\mathcal{GN}(v|m, s, \eta) = \frac{\eta}{2s\Gamma(\eta^{-1})}e^{-|r|^\eta}, \quad \text{where} \quad r := \frac{v-m}{s},$$

with location parameter $m \in \mathbb{R}$, scale parameter $s \in \mathbb{R}^+$ and shape parameter $\eta \in \mathbb{R}^+$. The Gamma function is defined as $\Gamma(x) := \int_0^\infty e^{-t}t^{x-1}dt$.

Moments. The generalised-normal distribution has moments: $\langle v \rangle = m$,

$$\mathrm{var}(v) = \frac{s^2\Gamma\left(3\eta^{-1}\right)}{\Gamma\left(\eta^{-1}\right)}, \quad \mathrm{skw}(v) = 0, \quad \text{and} \quad \mathrm{kur}(v) = \frac{\Gamma(5\eta^{-1})\Gamma(\eta^{-1})}{\Gamma(3\eta^{-1})^2} - 3.$$

Derivatives. The derivative of the log of the generalised-normal density with respect to the shape parameter η is

$$\frac{\partial}{\partial \eta} \log \mathcal{GN}\left(v|m,s,\eta\right) = \frac{1}{\eta} + \frac{1}{\eta^2} g\left(\frac{1}{\eta}\right) - |r|^{\eta} \log\left(|r|\right).$$

For affine independent KL inference we constrain $\eta > 1$ to ensure differentiability of the KL bound. We do not require the derivatives of the base density with respect to either the scale parameter or the location parameter since under the affine transformation these aspects of the density $q_{\mathbf{w}}(\mathbf{w})$ are already parameterised by $\mathbf{A}, \mathbf{b}$.

Entropy. The generalised-normal admits the following analytic form for the differential entropy

$$H\left[\mathcal{GN}\left(v|m,s,\eta\right)\right] = \frac{1}{\eta} - \log\left[\frac{\eta}{2s\Gamma\left(\eta^{-1}\right)}\right],$$

whose gradient with respect to η is given by

$$\frac{\partial}{\partial \eta} H\left[v\right] = -\frac{1}{\eta^2} - \frac{1}{\eta} + \text{digamma}\left(\frac{1}{\eta}\right)\frac{1}{\eta^2},$$

where the digamma function is defined as $\text{digamma}(x) = \frac{d}{dx}\log\Gamma(x)$.

Bibliography

J. Albert and S. Chib. Bayesian Analysis of Binary and Polychotomous Response Data. *Journal of the American Statistical Association*, 88(422):669–679, 1993. 30

S. Amari. Natural Gradient Works Efficiently in Learning. *Neural Computation*, 10(2):251–276, 1998. 39

C. Andrieu, N. De Freitas, A. Doucet, and M. Jordan. An Introduction to MCMC for Machine Learning. *Machine Learning*, 50(1-2):5–43, 2003. 34

D. Barber. *Bayesian Reasoning and Machine Learning*. Cambridge University Press, 2012. 23, 26, 35

D. Barber and C. Bishop. Ensemble Learning for Multi-Layer Networks. In *Advances in Neural Information Processing Systems, NIPS 10*, 1998a. 37, 48, 49, 58

D. Barber and C. Bishop. Ensemble Learning in Bayesian Neural Networks. In *Neural Networks and Machine Learning*, pages 215–237. Springer, 1998b. 12, 113

M. Beal. *Variational Algorithms for Approximate Bayesian Inference*. PhD thesis, University of London, 2003. 37

A. Bell and T. Sejnowski. Learning the Higher-Order Structure of a Natural Sound. *Network: Computation in Neural Systems*, 7(2):261–266, 1996. 30

J. Berger. *Statistical Decision Theory and Bayesian Analysis*. Springer Verlag, 1985. 21

D. Bickson and C. Guestrin. Inference with Multivariate Heavy-Tails in Linear Models. In *Advances in Neural Information Processing Systems, NIPS 23*, 2010. 97

C. Bishop, N. Lawrence, T. Jaakkola, and M. Jordan. Approximating Posterior Distributions in Belief Networks Using Mixtures. In *Advances in Neural Information Processing Systems, NIPS 10*, 1998. 55, 94, 101

D. Blei, M. Jordan, and J. Paisley. Variational Bayesian Inference with Stochastic Search. In *International Conference on Machine Learning, ICML-29*, 2012. 71

G. Bouchard and O. Zoeter. Split Variational Inference. In *International Conference on Artificial Intelligence and Statistics, AISTATS 12*, 2009. 56, 94, 101

S. Boyd and L. Vandenberghe. *Convex Optimization*. Cambridge University Press, 2004. 41, 60, 61, 120, 129, 130, 131, 134

S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein. Distributed Optimization and Statistical Learning via the Alternating Direction Method of Multipliers. *Foundations and Trends® in Machine Learning*, 3(1):1–122, 2011. 73

R. Bracewell. *The Fourier Transform and its Applications*. McGraw-Hill New York, 1986. 100

E. Challis and D. Barber. Concave Gaussian Variational Approximations for Inference in Large-Scale Bayesian Linear Models. In *International Conference on Artificial Intelligence and Statistics, AISTATS 14*, 2011. 12, 60, 124, 133

E. Challis and D. Barber. Affine Independent Variational Inference. In *Advances in Neural Information Processing Systems, NIPS 25*, 2012. 13

E. Challis and D. Barber. Gaussian Kullback-Leibler Approximate Inference. *The Journal of Machine Learning Research*, In press 2013. 13

K. Chaloner and I. Verdinelli. Bayesian Experimental Design: A Review. *Statistical Science*, pages 273–304, 1995. 22

O. Chapelle and L. Li. An Empirical Evaluation of Thompson Sampling. In *Neural Information Processing Systems, NIPS 25*, 2011. 72

T. Cover and J. Thomas. *Elements of Information Theory*. Wiley, New York, 1991. 36, 39, 96, 110

M. Cowles and B. Carlin. Markov Chain Monte Carlo Convergence Diagnostics: a Comparative Review. *Journal of the American Statistical Association*, 91(434):883–904, 1996. 34

L. Csató, E. Fokoué, M. Opper, B. Schottky, and O. Winther. Efficient Approaches to Gaussian Process Classification. In *Advances in Neural Information Processing Systems, NIPS 14*, 2000. 42, 122

B. Cseke and T. Heskes. Improving Posterior Marginal Approximations in Latent Gaussian Models. In *International Conference on Artificial Intelligence and Statistics, AISTATS 13*, 2010. 33

B. Cseke and T. Heskes. Approximate Marginals in Latent Gaussian Models. *The Journal of Machine Learning Research*, 12:417–454, 2011. 33

A. Dempster, N. Laird, and D. Rubin. Maximum Likelihood from Incomplete Data via the EM Algorithm. *Journal of the Royal Statistical Society. Series B*, pages 1–38, 1977. 25, 116

T. Eltoft, T. Kim, and T. Lee. On the Multivariate Laplace Distribution. *Signal Processing Letters, IEEE*, 13(5):300–303, 2006a. 73

T Eltoft, T. Kim, and T. Lee. Multivariate Scale Mixture of Gaussians Modeling. In *Independent Component Analysis and Blind Signal Separation*, pages 799–806. Springer, 2006b. 73

R. Fergus, B. Singh, A. Hertzmann, S. Roweis, and W. Freeman. Removing Camera Shake from a Single Photograph. In *ACM Transactions on Graphics*, 2006. 31, 85

J. Ferreira and M. Steel. A New Class of Skewed Multivariate Distributions with Applications To Regression Analysis. *Statistica Sinica*, 17:505–529, 2007. 95

J. Friedman. Multivariate Adaptive Regression Splines. *The Annals of Statistics*, 19(1):1–67, 1991. 79

G. Gershman, M. Hoffman, and D. Blei. Nonparametric Variational Inference. In *International Conference on Machine Learning, ICML 29*, 2012. 56, 94, 101

Z. Ghahramani and G. Hinton. The EM Algorithm for Mixtures of Factor Analyzers. Technical report, University of Toronto, CRG-TR-96-1, 1996. 23

M. Gibbs and D. MacKay. Variational Gaussian Process Classifiers. *IEEE Transactions on Neural Networks*, 11(6):1458–1464, 2000. 51

M. Girolami. A Variational Method for Learning Sparse and Overcomplete Representations. *Neural Computation*, 13(11):2517–2532, 2001. 11, 31, 51, 85

G. Golub and C. Van Loan. *Matrix Computations*. John Hopkins University Press, 1996. 45, 120, 129

C. Gourieroux and A. Monfort. *Statistics and Econometric Models*, volume 2. Cambridge University Press, 1995. 39

T. Graepel, J. Candela, T. Borchert, and R. Herbrich. Web-scale Bayesian Click-through Rate Prediction for Sponsored Search Advertising in Microsofts Bing Search Engine. In *International Conference on Machine Learning, ICML 27*, 2010. 72

A. Graves. Practical Variational Inference for Neural Networks. In *Advances in Neural Information Processing Systems, NIPS 24*, 2011. 69, 71

R. Herbrich. On Gaussian Expectation Propagation. Technical report, Microsoft Research Cambridge, 2005. Unpublished research note `research.microsoft.com/pubs/74554/EP.pdf`. 114, 130

G. Hinton and D. Van Camp. Keeping the Neural Networks Simple by Minimizing the Description Length of the Weights. In *Conference on Computational Learning Theory, COLT 6*, 1993. 12, 48

P. Højen-Sørensen, O. Winther, and L. Hansen. Mean-Field Approaches to Independent Component Analysis. *Neural Computation*, 14(4):889–918, 2002. 42

A. Honkela and H. Valpola. Unsupervised Variational Bayesian Learning of Nonlinear Models. In *Advances in Neural Information Processing Systems, NIPS 17*, 2005. 48

A. Honkela, T. Raiko, M. Kuusela, M. Tornio, and J. Karhunen. Approximate Riemannian Conjugate Gradient Learning for Fixed-Form Variational Bayes. *The Journal of Machine Learning Research*, 11: 3235–3268, Dec. 2010. 39, 69, 72

K. Hyun-Chul and Z. Ghahramani. Bayesian Gaussian Process Classification with the EM-EP Algorithm. In *IEEE Transactions on Pattern Analysis and Machine Intelligence 28*, 2006. 120

T. Jaakkola and M. Jordan. A Variational Approach to Bayesian Logistic Regression Problems and their Extensions. In *International Conference on Artificial Intelligence and Statistics, AISTATS 6*, 1997. 11, 51

M. Jamshidian and R. Jennrich. Conjugate Gradient Acceleration of the EM Algorithm. *Journal of the American Statistical Association*, 88(421):221–228, 1993. 115

P. Jylanki, J. Vanhatalo, and A. Vehtrari. Robust Gaussian Process Regression with a Student-t Likelihood. *The Journal of Machine Learning Research*, 12:3187–3225, 2011. 77, 79

E. Khan, S. Mohamed, and K. Murphy. Fast Bayesian Inference for Non-Conjugate Gaussian Process Regression. In *Advances in Neural Information Processing Systems, NIPS 25*, 2012. 64

M. Khan, B. Marlin, G. Bouchard, and K. Murphy. Variational Bounds for Mixed-Data Factor Analysis. In *Advances in Neural Information Processing Systems, NIPS 23*, 2010. 30, 31, 71, 100

M. Khan, A. Aravkin, M. Friedlander, and M. Seeger. Fast Dual Variational Inference for Non-Conjugate Latent Gaussian Models. In *International Conference on Machine Learning, ICML 30*, 2013. 73

H. Kim and Z. Ghahramani. Bayesian Gaussian Process Classification with the EM-EP Algorithm. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 28(12):1948–1959, 2006. 48

K. Ko and M. Seeger. Large Scale Variational Bayesian Inference for Structured Scale Mixture Models. In *International Conference on Machine Learning, ICML 29*, 2012. 53

M. Kuss. *Gaussian Process Models for Robust Regression, Classification, and Reinforcement Learning*. PhD thesis, Technischen Universität Darmstadt, Darmstadt, Germany, 2006. 44, 79, 103

T. Lee, M. Lewicki, M. Girolami, and T. Sejnowski. Blind Source Separation of More Sources than Mixtures Using Overcomplete Representations. *Signal Processing Letters, IEEE*, 6(4):87–90, 1999. 31

Y. Lim and Y. Teh. Variational Bayesian Approach to Movie Rating Prediction. In *Proceedings of KDD Cup and Workshop*, 2007. 71

D. MacKay. Bayesian Interpolation. *Neural Computation*, 4(3):415–447, 1992. 21

D. MacKay. Probable Networks and Plausible Predictions - a Review of Practical Bayesian Methods for Supervised Neural Networks. *Network: Computation in Neural Systems*, 6(3):469–505, 1995. 21

D. MacKay and J. Oldfield. Generalization Error and the Number of Hidden Units in a Multilayer Perceptron. Technical report, Cambridge University, `www.inference.phy.cam.ac.uk/mackay/gen.ps.gz`, 1995. 83

D. MacKay, R. Turner, and M. Sahani. Compactness of Variational Approximations. Research note. www.inference.phy.cam.ac.uk/mackay/compact.pdf, 2008. 38

C. Manski. The Structure of Random Utility Models. *Theory and Decision*, 8(3):229–254, 1977. 30

B. Marlin, M. Khan, and K. Murphy. Piecewise Bounds for Estimating Bernoulli-Logistic Latent Gaussian Models. In *International Conference on Machine Learning, ICML 28*, 2011. 11, 130

P. McCullagh and J. Nelder. *Generalized Linear Models*. Chapman & Hall, 1989. 29

T. Minka. Automatic Choice of Dimensionality for PCA. In *Advances in Neural Information Processing Systems, NIPS 14*, 2000. 24

T. Minka. *A Family of Algorithms for Approximate Bayesian Inference*. PhD thesis, MIT Media Lab, 2001a. 38, 45, 114

T. Minka. Expectation Propagation for Approximate Bayesian Inference. In *Conference on Uncertainty in Artificial Intelligence, UAI 17*, 2001b. 45

T. Minka. Power EP. Technical report, Department of Statistics, Carnegie Mellon University, research.microsoft.com/pubs/67427/tr-2004-149.pdf, 2004. 47, 80, 114

T. Minka. Divergence Measures and Message Passing. Technical report, Microsoft Research Cambridge, Tech. Rep. MSR-TR-2005-173, ftp.research.microsoft.com/pub/tr/TR-2005-173.pdf, 2005. 36, 38

R. Neal. Monte Carlo Implementation of Gaussian Process Models for Bayesian Regression and Classification. Technical report, Department of Statistics and Department of Computer Science, University of Toronto, arxiv.org/abs/physics/9701026v2, 1997. 79

H. Nickisch. *Bayesian Inference and Experimental Design for Large Generalised Linear Models*. PhD thesis, Technische Universität Berlin, Berlin, Germany, 2010. 90

H. Nickisch. glm-ie: Generalised Linear Models Inference and Estimation Toolbox. *The Journal of Machine Learning Research*, 13:1699–1703, 13 2012. 82

H. Nickisch and C. Rasmussen. Approximations for Binary Gaussian Process Classification. *The Journal of Machine Learning Research*, 9:2035–2078, 10 2008. 48, 50, 53, 54, 67

H. Nickisch and M. Seeger. Convex Variational Bayesian Inference for Large Scale Generalized Linear Models. In *International Conference on Machine Learning, ICML 26*, 2009. 53, 69, 84

J. Nocedal and S. Wright. *Numerical Optimization*. Springer, 2006. 41, 61, 88

J. Nolan. *Stable Distributions - Models for Heavy Tailed Data*. Birkhauser, Boston, 2012. In progress, Chapter 1 online at academic2.american.edu/~jpnolan. 97

B. Olshausen and D. Field. Natural Image Statistics and Efficient Coding. *Network: Computation in Neural Systems*, 7:333–339, 2 1996. 30, 86

M. Opper. *Advanced Mean Field Methods: Theory and Practice*. MIT press, 2001. 37

M. Opper and C. Archambeau. The Variational Gaussian Approximation Revisited. *Neural Computation*, 21(3):786–792, 2009. 37, 48, 49, 50, 69, 77, 102

M. Opper and O. Winther. Expectation Consistent Approximate Inference. *The Journal of Machine Learning Research*, 6:2177–2204, Dec. 2005. 47

J. Ormerod. Skew-Normal Variational Approximations for Bayesian Inference. Technical Report CRG-TR-93-1, School of Mathematics and Statistics, University of Sydney, 2011. 56

J. Ormerod and M. Wand. Gaussian Variational Approximate Inference for Generalized Linear Mixed Models. *Journal of Computational and Graphical Statistics*, 21(1):2–17, 2012. 69

A. Palmer, D. Wipf, K. Kreutz-Delgado, and B. Rao. Variational EM Algorithms for non-Gaussian Latent Variable Models. In *Advances in Neural Information Processing Systems, NIPS 20*, 2006. 51, 52

G. Papandreou and A. Yuille. Gaussian Sampling by Local Perturbations. In *Advances in Neural Information Processing Systems, NIPS 11*, 2010. 53

T. Park and G. Casella. The Bayesian Lasso. *Journal of the American Statistical Association*, 103: 681–686, 2008. 11

C. Rasmussen and H. Nickisch. Gaussian Processes for Machine Learning (GPML) Toolbox. *The Journal of Machine Learning Research*, 11:3011–3015, Nov. 2010. 77

C. Rasmussen and C. Williams. *Gaussian Processes for Machine Learning*. MIT Press, 2006. 75

P. Ruckdeschel and M. Kohl. General Purpose Convolution Algorithm in S4-Classes by Means of FFT. Technical Report 1006.0764v2, arXiv.org, 2010. 100

H. Rue, S. Martino, and N. Chopin. Approximate Bayesian Inference for Latent Gaussian Models by Using Integrated Nested Laplace Approximations. *Journal of the royal statistical society. Series B*, 71 (2):319–392, 2009. 33

S. Sahu, D. Dey, and M. Branco. A New Class of Multivariate Skew Distributions with Applications to Bayesian Regression Models. *The Canadian Journal of Statistics / La Revue Canadienne de Statistique*, 31(2):129–150, 2003. 95

R. Salakhutdinov and A. Mnih. Bayesian Probabilistic Matrix Factorization using Markov Chain Monte Carlo. In *International Conference on Machine Learning, ICML 25*, 2008. 71

R. Salakhutdinov, S. Roweis, and Z. Ghahramani. Optimization with EM and Expectation-Conjugate-Gradient. In *International Conference on Machine Learning, ICML 20*, 2003. 91, 115

L. Saul, T. Jaakkola, and M. Jordan. Mean Field Theory for Sigmoid Belief Networks. *The Journal of Artificial Intelligence Research*, 4:61–76, 1996. 51

P. Schaller and G. Temnov. Efficient and Precise Computation of Convolutions: Applying FFT to Heavy Tailed Distributions. *Computational Methods in Applied Mathematics*, 8(2):187–200, 2008. 100, 104, 105

M. Schmidt, G. Fung, and R. Rosales. Fast Optimization Methods for L1 Regularization: A Comparative Study and Two New Approaches. In *European Conference on Machine Learning, ECML 18*, 2007. 61

M. Seeger. Bayesian Methods for Support Vector Machines and Gaussian Processes. Master's thesis, University of Karlsruhe, 1999. 49, 63

M. Seeger. Expectation Propagation for Exponential Families. Technical report, University of California at Berkeley, `www.kyb.tuebingen.mpg.de/bs/people/seeger`, 2005. 47

M. Seeger. Low Rank Updates for the Cholsky Decomposition. Technical report, University of California at Berkeley, `infoscience.epfl.ch/record/161468/files/cholupdate.pdf`, 2007. 52, 121

M. Seeger. Bayesian Inference and Optimal Design in the Sparse Linear Model. *The Journal of Machine Learning Research*, 9:759–813, Oct. 2008. 35, 44, 80

M. Seeger. Sparse Linear Models: Variational Approximate Inference and Bayesian Experimental Design. *Journal of Physics: Conference Series*, 197(1), 2009. 67, 86

M. Seeger. Gaussian Covariance and Scalable Variational Inference. In *International Conference on Machine Learning, ICML 27*, 2010. 45, 53, 129

M. Seeger and G. Bouchard. Fast Variational Bayesian Inference for Non-Conjugate Matrix Factorization Models. In *International Conference on Artificial Intelligence and Statistics, AISTATS 15*, 2012. 71

M. Seeger and H. Nickisch. Compressed Sensing and Bayesian Experimental Design. In *International Conference on Machine Learning, ICML 25*, 2008. 85, 86

M. Seeger and H. Nickisch. Large Scale Variational Inference and Experimental Design for Sparse Generalized Linear Models. Technical report, Max Planck Institute for Biological Cybernetics, `arxiv.org/abs/0810.0901`, 2010. 51, 66, 77

M. Seeger and H. Nickisch. Fast Convergent Algorithms for Expectation Propagation Approximate Inference. In *International Conference on Artificial Intelligence and Statistics, AISTATS 14*, 2011a. 47, 69

M. Seeger and H. Nickisch. Large Scale Bayesian Inference and Experimental Design for Sparse Linear Models. *SIAM Journal on Imaging Sciences*, 4(1):166–199, 2011b. 53, 61, 63, 80, 86, 87, 90

M. Seeger, S. Gerwinn, and M. Bethge. Bayesian Inference for Sparse Generalized Linear Models. In *European Conference on Machine Learning, ECML 18*, 2007. 47, 77

S. Seo, M. Wallat, T. Graepel, and K. Obermayer. Gaussian Process Regression: Active Data Selection and Test Point Rejection. In *Mustererkennung 2000*, pages 27–34. Springer, 2000. 22

J. Tenenbaum and W. Freeman. Separating Style and Content with Bilinear Models. *Neural Computation*, 12(6):1247–1283, 2000. 71

M. Tipping. Probabilistic Visualisation of High-dimensional Binary Data. In *Advances in Neural Information Processing Systems, NIPS 11*, 1999. 11, 30, 31

M. Tipping and C. Bishop. Probabilistic Principal Component Analysis. *Journal of the Royal Statistical Society. Series B*, 61(3):611–622, 1999. 23, 71

M. Titsias and M. Lázaro-Gredilla. Spike and Slab Variational Inference for Multi-Task and Multiple Kernel Learning. In *Advances in Neural Information Processing Systems, NIPS 24*, 2012. 42

J. Vanhatalo, P. Jylnki, and A. Vehtari. Gaussian Process Regression with a Student-t Likelihood. In *Advances in Neural Information Processing Systems, NIPS 22*, 2009. 11, 77, 79

M. Wainwright and M. Jordan. Graphical Models, Exponential Families, and Variational Inference. *Foundations and Trends in Machine Learning*, 1(1-2):1–305, 2008. 35

M. Welling, C. Chemudugunta, and N. Sutter. Deterministic Latent Variable Models and their Pitfalls. In *SIAM International Conference on Data Mining*, 2008. 41

D. Wipf. Sparse Bayesian Learning for Basis Selection. *IEEE Transactions on Signal Processing*, 52 (8):2153–2164, 2004. 85

N. Wu and J. Zhang. Factor-analysis Based Anomaly Detection and Clustering. *Decision Support Systems*, 42(1):375–389, 2006. 23

J. Zhao, L. Philip, and Q. Jiang. ML Estimation for Factor Analysis: EM or non-EM? *Statistics and Computing*, 18(2):109–123, 2008. 25

O. Zoeter and T. Heskes. Gaussian Quadrature Based Expectation Propagation. In *International Conference on Artificial Intelligence and Statistics, AISTATS 10*, 2005. 115