Bayesian Inference and Experimental Design for Large Generalised Linear Models

vorgelegt von

Dipl.-Inf. Hannes Nickisch

aus Leipzig

Von der Fakultät IV - Elektrotechnik und Informatik

der Technischen Universität Berlin

zur Erlangung des akademischen Grades

Doktor der Naturwissenschaften

– Dr. rer. nat. –

genehmigte Dissertation.

Promotionsausschuss:

Vorsitzender:	Prof. Dr. Klaus-Robert Müller
Berichter:	Prof. Dr. Manfred Opper
Berichter:	PhD. Matthias W. Seeger
Berichter:	PhD. Carl E. Rasmussen
Sachverständiger:	Prof. Dr. Klaus Obermayer
Sachverständiger:	Prof. Dr. Felix Wichmann

Tag der wissenschaftlichen Aussprache: 17. September 2010

Berlin 2010
D83

Acknowledgements

The most important ingredient for this work was the open, cordial and scientific atmosphere at the Empirical Inference Department of the Max Planck Institute for Biological Cybernetics, Tübingen. I am thankful to Bernhard Schölkopf for creating this stimulating as well as productive environment and giving input and feedback whenever necessary.

Thanks to Carl Rasmussen, I learned about Gaussian processes, approximate inference, differentiating between the "big picture" and "technical details" and concise scientific programming, experimenting and writing.

Matthias Seeger's enthusiasm, experience in probabilistic modelling and knowledge on mathematics and numerical optimisation where a constant driving force and source of ideas. Without his supervision, this thesis wouldn't have been possible.

Manfred Opper provided formal supervision, valuable discussions and convivial receptions in Berlin.

Last but not least, I would like to thank my family; especially Susanne and Oskar for enduring and also mentally supporting the writing and the defence.

Zusammenfassung

Zu Entscheidungen zu gelangen trotz unsicherer und unvollständiger Informationen, ist eines der zentralen Themen der Statistik und des maschinellen Lernens. Probabilistische Bayesianische Modelle stellen dabei einen strengen mathematischen Rahmen für die Formalisierung der Datengewinnung zur Verfügung, in dem getroffene Annahmen sowie vorhandenes Vorwissen explizit gemacht werden. Die resultierende a-posteriori-Verteilung repräsentiert den Wissensstand des Modells und ist Ausgangspunkt für sich anschließende Entscheidungen.

Trotz aller begrifflichen Klarheit der Bayesianischen Inferenz haben die notwendigen Berechnungen meist die Form analytisch unlösbarer hochdimensionaler Integrale, was in der Praxis zu einer Reihe von randomisierten und deterministischen Näherungsverfahren führt.

Die vorliegende Arbeit entwickelt, studiert und wendet Algorithmen zur näherungsweisen Inferenz und Versuchsplanung auf generalisierte lineare Modelle (GLM) an. Ein besonderer Schwerpunkt liegt auf algorithmischen Eigenschaften wie Konvexität, numerische Stabilität und Skalierbarkeit hin zu großen Mengen an wechselwirkenden Größen.

Nach einer Einführung in GLMs stellen wir die vielversprechendsten Ansätze zum Schätzen, zur näherungsweisen Inferenz und zur Versuchsplanung vor.

Wir untersuchen detailliert einen speziellen Ansatz und leiten Konvexitäts-Eigenschaften her, was zu einem generischen und skalierbaren Inferenzverfahren führt. Desweiteren sind wir in der Lage, den Zusammenhang zwischen Bayesianischer Inferenz und dem regularisierten statistischen Schätzen genau zu beschreiben: Schätzen ist ein Spezialfall von Inferenz und Inferenz kann durch eine Folge von geglätteten Schätzern berechnet werden.

Im Anschluss daran vergleichen wir eine Reihe von Inferenzverfahren, angewendet auf die binäre probabilistische Klassifikation mittels eines kernbasierten GLMs, dem sogenannten Gauß-Prozess-Modell. Eine Reihe empirischer Experimente ermittelt den EP-Algorithmus als das genaueste Näherungsverfahren.

In einem nächsten Schritt wenden wir den EP-Algorithmus auf die sequenzielle Optimierung der Messarchitektur eines Bilderfassungssystems an. Dies unter Verwendung von Compressive Sampling (CS), bei dem die intrinsische Redundanz in Signalen benutzt wird, um den Messprozess zu beschleunigen. In vergleichenden Experimenten beobachten wir Unterschiede zwischen dem Verhalten von adaptivem CS in der Praxis und dem theoretisch untersuchten Szenario.

Durch Kombination der gewonnenen Erkenntnisse über adaptives CS mit unserem konvexen Inferenzverfahren sind wir in der Lage, die Messsequenz von Magnetresonanztomographie-Systemen (MRT) zu verbessern, indem wir das Bayesianische Kriterium zur Versuchsplanung optimieren. Unsere MRT-Anwendung auf Bildern realitischer Größe ermöglicht kürzere Messzeiten bei gleichbleibender Bildqualität.

Abstract

Decision making in light of uncertain and incomplete knowledge is one of the central themes in statistics and machine learning. Probabilistic Bayesian models provide a mathematically rigorous framework to formalise the data acquisition process while making explicit all relevant prior knowledge and assumptions. The resulting posterior distribution represents the state of knowledge of the model and serves as the basis for subsequent decisions.

Despite its conceptual clarity, Bayesian inference computations take the form of analytically intractable high-dimensional integrals in practise giving rise to a number of randomised and deterministic approximation techniques.

This thesis derives, studies and applies deterministic approximate inference and experimental design algorithms with a focus on the class of generalised linear models (GLMs). Special emphasis is given to algorithmic properties such as convexity, numerical stability, and scalability to large numbers of interacting variables.

After a review of the relevant background on GLMs, we introduce the most promising approaches to estimation, approximate inference and experiment design.

We study in depth a particular approach and reveal its convexity properties naturally leading to a generic and scalable inference algorithm. Furthermore, we are able to precisely characterise the relationship between Bayesian inference and penalised estimation: estimation is a special case of inference and inference can be done by a sequence of smoothed estimation steps.

We then compare a large body of inference algorithms on the task of probabilistic binary classification using a kernelised GLM: the Gaussian process model. Multiple empirical comparisons identify expectation propagation (EP) as the most accurate algorithm.

As a next step, we apply EP to adaptively and sequentially design the measurement architecture for the acquisition of natural images in the context of compressive sensing (CS), where redundancy in signals is exploited to accelerate the measurement process. We observe in comparative experiments differences between adaptive CS results in practise and the setting studied in theory.

Combining the insights from adaptive CS with our convex variational inference algorithm, we are able – by sequentially optimising Bayesian design scores – to improve the measurement sequence in magnetic resonance imaging (MRI). In our MRI application on realistic image sizes, we achieve scan time reductions for constant image quality.

Contents

List of Figures

List of Algorithms

List of Tables

Notation

Matrices, vectors and scalars

$\mathbf{x}, \boldsymbol{\gamma}$	Bold lower case letters denote column vectors
$\mathbf{X}, \boldsymbol{\Gamma}$	Bold upper case letters denote matrices
$\mathbf{x}^j$	The j-th column vector $\mathbf{x}$ of a matrix $\mathbf{X}$
x_i	The i-th element of a vector $\mathbf{x}$
x_i^j	The i-th element of a vector $\mathbf{x}^j$
X_{ij}	The ij-th element of a matrix $\mathbf{X}$
$\mathbf{x} = [x_i]_i$	Formation of a vector $\mathbf{x}$ from scalars x_i
$\mathbf{X} = [X_{ij}]_{ij}$	Formation of a matrix $\mathbf{X}$ from scalars X_{ij}
$\mathbf{1}$	Vector of ones
$\mathbf{I}$	Identity matrix, $\mathbf{I} = \mathrm{dg}(\mathbf{1})$
$\mathbf{e}_i$	i-th unit vector, $\mathbf{I} = [\mathbf{e}_1, .., \mathbf{e}_n]$
$[\mathbf{A}, \mathbf{B}]$	horizontal matrix concatenation along rows
$[\mathbf{A}; \mathbf{B}] = [\mathbf{A}^\top, \mathbf{B}^\top]^\top$	vertical matrix concatenation along columns

Operations and relations

$\mathbf{a} \odot \mathbf{b}, \mathbf{A} \odot \mathbf{B}$	Hadamard point wise product between vectors or matrices
$a \cdot \mathbf{B}, a \cdot \mathbf{b}, a \cdot b$	Multiplication with a scalar a (to explicitly highlight it)
$\mathbf{a}^n$	Vector component wise power $\mathbf{a}^n = \mathbf{a} \odot \mathbf{a} \odot ... \odot \mathbf{a}$
$\mathbf{A}^n$	Matrix power $\mathbf{A}^n = \mathbf{A}\mathbf{A}...\mathbf{A}$
$\mathrm{dg}(\mathbf{a}) = \mathbf{A}$	Diagonal matrix $\mathbf{A}$ with diagonal $\mathbf{a}$
$\mathbf{a} = \mathrm{dg}(\mathbf{A})$	$\mathbf{a}$ is the diagonal of the matrix $\mathbf{A}$
$\mathbf{a}^\top, \mathbf{X}^\top$	Vector or matrix transpose
$\mathbf{a}^\mathsf{H}, \mathbf{X}^\mathsf{H}$	Conjugate transpose of a complex vector or matrix $\mathbf{X}^\mathsf{H} = \overline{\mathbf{X}}^\top$
$\mathbf{A}^{-1}$	Matrix inverse, $\mathbf{A}^{-1}\mathbf{A} = \mathbf{I}$
$\mathbf{A}^{-\top}, \mathbf{A}^{-\mathsf{H}}$	Matrix inverse transpose, $\mathbf{A}^{-\top} = (\mathbf{A}^{-1})^\top$, $\mathbf{A}^{-\mathsf{H}} = (\mathbf{A}^{-1})^\mathsf{H}$
$\mathbf{X}^+$	Pseudo inverse, $\mathbf{X}^+ = \lim_{\delta \to 0} \mathbf{X}^\top(\mathbf{X}\mathbf{X}^\top + \delta\mathbf{I})^{-1} = \lim_{\delta \to 0}(\mathbf{X}^\top\mathbf{X} + \delta\mathbf{I})^{-1}\mathbf{X}^\top$
$\mathrm{tr}(\mathbf{A}) = \mathbf{1}^\top \mathrm{dg}(\mathbf{A})$	Trace of $\mathbf{A}$, sum of entries on the diagonal
$\lvert\mathbf{A}\rvert$	Determinant
$\mathbf{a} \succeq \mathbf{b}, \mathbf{A} \succeq \mathbf{B}$	Component-wise relation, $a_i \geq b_i$, $A_{ij} \geq B_{ij}$
$\mathbf{A} \succcurlyeq \mathbf{B}$	Full matrix relation, $\mathbf{A} - \mathbf{B}$ is positive semidefinite
$\overset{c}{=}, \overset{c}{\geq}, \overset{c}{\approx}$	Relation up to a constant
$f(\mathbf{x}) \propto g(\mathbf{x})$	Proportionality, $\exists \alpha \forall \mathbf{x} : \; f(\mathbf{x}) = \alpha g(\mathbf{x})$

Complex numbers

$z = a + b\mathrm{i} = re^{\mathrm{i}\varphi}$	A complex number, $\mathrm{i}^2 = -1$
$a = \Re(z)$, $b = \Im(z)$	Cartesian form: real and imaginary part
$r = \lvert z \rvert$, $\varphi = \measuredangle(z)$	Polar/trigonometric form: absolute value and phase/angle
$\overline{a + b\mathrm{i}} = a - b\mathrm{i}$	Conjugation

Functions

$f(\cdot)$, $k(a, \cdot)$	A function in one argument
$x \mapsto y$	An anonymous function
$f : \mathcal{X} \to \mathcal{Y}$	Domain and codomain specification for a function
$f \circ g$	Function concatenation, $(f \circ g)(x) = f(g(x))$

Derivatives

$\frac{\partial f(\mathbf{x})}{\partial \mathbf{x}} \equiv \frac{\partial f}{\partial \mathbf{x}}$	Vector of partial derivatives of f w.r.t. $\mathbf{x}$
$\frac{\partial^2 f}{\partial \mathbf{x} \partial \mathbf{x}^\top}$	Hessian matrix of second partial derivatives of f w.r.t. $\mathbf{x}$
$\frac{\partial \mathbf{f}}{\partial \mathbf{x}^\top}$	Matrix of partial derivatives of $\mathbf{f}$ w.r.t. $\mathbf{x}$
$\mathrm{d}\mathbf{x}$	The differential of $\mathbf{x}$
$\frac{\mathrm{d}f(\mathbf{x},\mathbf{z})}{\mathrm{d}\mathbf{x}} \equiv \frac{\mathrm{d}f}{\mathrm{d}\mathbf{x}}$	The vector of total derivatives of f w.r.t. $\mathbf{x}$
$\frac{\delta F(f)}{\delta f}$	The functional derivative of F w.r.t. the function f
$\nabla f = \frac{\partial f}{\partial \mathbf{x}}$, $\nabla_{\mathbf{x}} f$	The gradient of $f(\mathbf{x})$ (at $\mathbf{x}$)

Probability

$\mathbb{P}(\mathbf{x})$	Probability density function over $\mathbf{x}$
$\mathbb{Q}(\mathbf{x})$	An approximation to $\mathbb{P}(\mathbf{x})$
$\mathbb{E}\left[f(\mathbf{x})\right], \mathbb{E}_{\mathbb{P}(\mathbf{x})}\left[f(\mathbf{x})\right]$	Expectation of $f(\mathbf{x})$, $\mathbb{E}\left[\mathbf{x}\right] = \int f(\mathbf{x})\mathbb{P}(\mathbf{x})\mathrm{d}\mathbf{x}$
$\mathbb{V}\left[\mathbf{x}\right]$	Covariance of $\mathbf{x}$, $\mathbb{V}\left[\mathbf{x}\right] = \mathbb{E}\left[\left(\mathbf{x} - \mathbb{E}\left[\mathbf{x}\right]\right)\left(\mathbf{x} - \mathbb{E}\left[\mathbf{x}\right]\right)^\top\right]$
$\mathcal{H}\left[\mathbb{P}\right]$ or $\mathcal{H}\left[\mathbf{x}\right]$	Entropy of $\mathbb{P}$ or $\mathbf{x} \sim \mathbb{P}(\mathbf{x})$, $\mathcal{H}\left[\mathbb{P}\right] = \mathbb{E}\left[-\ln \mathbb{P}(\mathbf{x})\right]$
$\mathcal{I}(x_i, x_j) \geq 0$	Mutual information, $\mathcal{I}(x_i, x_j) = \mathbb{E}_{\mathbb{P}(x_i,x_j)}\left[-\ln\left(\mathbb{P}(x_i)\mathbb{P}(x_j)\right)\right] - \mathcal{H}\left[\mathbb{P}(x_i, x_j)\right]$
$\mathrm{KL}(\mathbb{Q} \vert\vert \mathbb{P})$	Kullback-Leibler divergence, $\mathrm{KL}(\mathbb{Q} \vert\vert \mathbb{P}) = \mathbb{E}\left[-\ln \mathbb{Q}(\mathbf{x})\right] - \mathcal{H}\left[\mathbb{Q}\right]$
$D_\alpha(\mathbb{P} \vert\vert \mathbb{Q})$	Alpha divergence, $D_\alpha(\mathbb{P} \vert\vert \mathbb{Q}) = \frac{1}{\alpha(1-\alpha)}(1 - \int \left[\mathbb{P}(\mathbf{x})\right]^\alpha \left[\mathbb{Q}(\mathbf{x})\right]^{1-\alpha} \mathrm{d}\mathbf{x})$
$\mathcal{N}(\mathbf{x} \vert \boldsymbol{\mu}, \boldsymbol{\Sigma})$	Multivariate Gaussian, $\mathcal{N}(\mathbf{x} \vert \boldsymbol{\mu}, \boldsymbol{\Sigma}) = \lvert 2\pi\boldsymbol{\Sigma} \rvert^{-\frac{1}{2}} \exp\left(-\frac{1}{2}(\mathbf{x} - \boldsymbol{\mu})^\top \boldsymbol{\Sigma}^{-1}(\mathbf{x} - \boldsymbol{\mu})\right)$
$\delta(\mathbf{x})$	Dirac delta distribution, e.g. $\delta(\mathbf{x}) = \lim_{\epsilon \to 0} \mathcal{N}(\mathbf{x} \vert \mathbf{0}, \epsilon\mathbf{I})$

Chapter 1

Introduction

Science in general constructs models of the world from incomplete, uncertain and possibly irrelevant or redundant information. Models or theories are built from particular experience or experiments, but are intended to explain or predict general circumstances. Different fields such as statistical science, mathematics and philosophy study the principle of inductive reasoning or equivalently, the adaptation of a process model in the light of observed data from a physical process. Growing computational resources lead to the emergence of *machine learning,* where adaptive and predictive systems are both theoretically and empirically studied. Machine learning, as an empirical science, is a "loose confederation of themes in statistical inference and decision making" [Jordan, 2009] with a focus on exploratory data analysis and computational methodology. Strong ties to signal processing, linear algebra, and optimisation make machine learning an interdisciplinary field trying to understand, apply and improve predictive models developed in statistics, computer science and engineering.

1.1 Learning

Machine learning – being an important, active, modern and successful branch of artificial intelligence research – is concerned with the design of algorithms enabling machines to *learn*. Learning is understood as automatic extraction of general rules about the population from a small sample in order to make predictions and decisions. The term in statistics most equivalent to learning is *estimation*. Computer scientists talk about soft computing. Learning cannot be successful without any prior assumption on the regularity of the underlying mechanisms. The goal of a researcher in machine learning is therefore to make as little assumptions as possible and to make them as explicit as possible. One of the central challenges in machine learning is the balance between learning and memorising, i.e. the trade-off between the general rule and the particular data.

Prior assumptions According to the no-free-lunch theorems [Wolpert, 1996], all learning algorithms perform equally well if averaged over all possible learning problems. Thus, prior knowledge or prior assumptions on the particular problem at hand like smoothness of the underlying function are indispensable for successful learning.

Available structure Since data is digitised and represented on a computer, every single datum is described by a collection of numbers. Inference becomes possible due to *additional structure* in these numbers. Mutual dependencies, exclusive constraints or simply redundancy enables predictions – it is impossible to predict one quantity from an independent other quantity. Often this structure is congruent with mathematical objects like sets, graphs or vector spaces allowing for operations such as addition and scaling. Besides the structure inside a single data point, also relations between data points facilitate predictions. They can be formalised by concepts like *similarity*, *distance* or *covariance* that establish pairwise relationships. Based on pairwise relations one can (at least approximately) embed data into linear spaces and exploit their favourable analytical properties.

Feedback Learning from examples can be done in different settings, the simplest being *supervised learning*, where the target label is provided for every training point. Labelling is laborious; hence *semi-supervised learning* uses unlabelled examples to improve predictions. *Unsupervised learning* directly identifies relevant structure in the dataset itself, without any labelling given, which helps is compression and visualisation. A weak form of feedback is used in *reinforcement learning*, where targets are not provided explicitly, but a series of decisions is judged as a whole.

1.2 Probabilistic models

Real data is incomplete, redundant, noisy and partially irrelevant to the problem at hand, models are only abstractions necessarily neglecting some details and even the world itself is not deterministic. Therefore, any machine learning algorithm has to deal with uncertainties originating from various sources. A proper treatment of uncertainty includes representation, updating and quantification of confidence in light of a prediction task.

Coping with noise by designing robust algorithms that are insensitive to small changes of data or large changes of a tiny fraction of data is dangerous. This deterministic approach ignores possibly relevant structure. For a principled approach to design of predictive models, explicit inclusion of noise models is indispensable.

The language of probability theory has many advantages. First of all, it is a natural way to describe ignorance or missing knowledge. Second, all assumptions have to made explicit since the calculus of probabilities is incorruptible – only a fully specified model allows proper inference. And third, probabilistic models have a standardised and normalised interface to the outside world facilitating composition of systems: a probability. Thus, if hierarchical models are to be built or sequences of designs have to be made, there is no way around probabilistic models.

Unfortunately, heavy computational challenges due to high-dimensional integrals lurk behind the formal beauty of a fully probabilistic model. There are two ways around: either, approximations are inevitable or tractable models have to be used. In practice, the by far most tractable multivariate distribution over continuous variables is the Gaussian distribution. Computations with Gaussians reduce to linear algebra, which makes them tractable in high dimensions. Sums of many random variables behave like a Gaussian, the Gaussian is the least structured density – there is a long list of favourable properties making the Gaussian family the major working horse in approximate inference. One central idea of this thesis is to make strong use of the "Gaussian toolbox": Gaussian approximations, Gaussian distributions, Gaussian scale mixtures, Gaussian processes, Gaussian quadrature, and the Gauss-Newton algorithm etc. in order to deal with high-dimensional integrals in Bayesian inference.

On the other hand, a model should be as simple as possible. Therefore, modelling a high-dimensional density might be a waste of resources if only a single decision is the goal of the analysis. Direct, not necessarily probabilistically motivated prediction models might work as well.

The axioms of probability and the induced calculus are mathematically not debatable. However, people interpret probabilities differently: either as relative frequencies of many repetitions of the same experiment or as a belief reflecting the lack of knowledge of the current state of nature. Feeling that too much paper and ink have been wasted to only insist on the differences, we want to stress the complementary but not exclusive nature of the Bayesian and frequentist points of view and rather think of them as displaying their respective strengths in different application settings as detailed in chapter 2.1.

1.3 Summary of the contributions

The thesis at hand includes theoretical, empirical and algorithmical insights. Further it puts material and ideas into perspective and provides code. Core machine learning techniques are applied to image acquisition and medical imaging. The basic theme is the generic goal to render Bayesian analysis feasible via approximate algorithms exploiting standard techniques from numerical mathematics, signal processing and optimisation while staying as generic and scalable as possible.

The study of convexity properties of variational inference as detailed in chapter 3 is a theoretical contribution. The scalable double loop algorithm from chapter 3 and its application to magnetic resonance imaging in chapter 6 as well as the ideas about how to run expectation propagation efficiently on a medium scale in a sequential fashion of chapter 5 are part of the algorithmical contribution. Our finding that a simple measurement heuristic shows clear advantages over randomised acquisition in chapter 5 suggests that second order structure is underrepresented in theoretical research on compressive sampling. Empirical observations and comparisons of approximate inference techniques are given in chapter 4. Finally, we can conclude that the Bayesian method bears advantages if used for subsequent experimental design, where a correct quantification of uncertainty is needed.

1.4 Outline of the thesis

The thesis comprises an introductory chapter (1), a chapter discussing the basics of statistical inference (2), four technical chapters (3, 4, 5, 6) and a final chapter providing a summary (7). The chapter dependency DAG (directed acyclic graph) is given below.

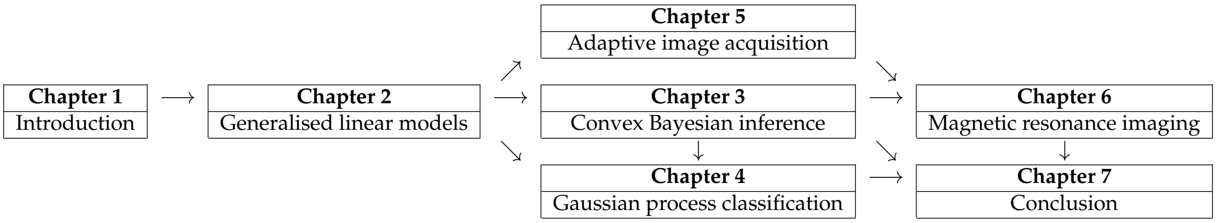

Chapter 1
Introduction
Chapter 2
Generalised linear models
Chapter 5
Adaptive image acquisition
Chapter 3
Convex Bayesian inference
Chapter 4
Gaussian process classification
Chapter 6
Magnetic resonance imaging
Chapter 7
Conclusion

Table 1.1: Thesis chapter dependency graph

After a review of the most prominent approximation techniques for Bayesian inference in continuous models in chapter 2, chapter 3 gives a characterisation of the convexity properties of a particular relaxation to variational inference along with a scalable algorithm. Subsequently, chapter 4 applies the framework to probabilistic classification and provides empirical insights into the behaviour of the inference procedures in practise; expectation propagation being the most accurate one. The following two chapters apply the experimental design methodology to image acquisition: first, we show in chapter 5, how to operate the expectation propagation machinery in the regime of a few thousand data points and empirically demonstrate the advantages of adaptive compressive sampling over random designs. Second, we scale the model of chapter 5 to realistic image sizes and employ the algorithm of chapter 3 for inference. In chapter 6, we describe the resulting feasible offline optimisation scheme that allows adjusting the magnetic resonance image acquisition process in a data driven way. As a result, we are able to not only reconstruct images from undersampled measurements but to sequentially select the measurements to make the undersampled reconstruction as faithful as possible.

1.5 Publication record

Most of the material of this thesis is already published, only parts are currently under review for publication. The study about approximate inference schemes for binary Gaussian process classification [Nickisch and Rasmussen, 2008] and the associated code [Rasmussen and Nickisch, 2010] is presented in chapter 4, the application of Bayesian experimental design to compressive sensing of natural images [Seeger and Nickisch, 2008a] is included in chapter 5. Chapter 3 introduces a convex algorithm for large-scale inference [Nickisch and Seeger, 2009, Seeger and Nickisch, 2008b, 2010, submitted] and chapter 6 details the benefits of optimising the k-space trajectories for Magnetic Resonance Image acquisition as published in Seeger, Nickisch, Pohmann, and Schölkopf [2009] and Seeger, Nickisch, Pohmann, and Schölkopf [2010].

Some material from the domain of computer vision has been omitted because it does not thematically fit into the exposition. In particular, the approach to learn object detectors from an intermediate attribute layer rather than from simple features [Lampert, Nickisch, and Harmeling, 2009] is not included. We did not incorporate the training and test methodology for interactive image segmentation systems [Nickisch et al., accepted]. The project using Gaussian process latent variable models for density modelling [Nickisch and Rasmussen, in press] is not part of the thesis, as well.

Chapter 2

Inference and Design in Linear Models

Suppose we are given a vector of *observations* $\mathbf{y} = [y_1, .., y_m]^\top$ with corresponding *covariates* or *data* $\mathbf{X} = [\mathbf{x}_1, .., \mathbf{x}_m]^\top$ and we wish to model the functional relationship $f : \mathbf{x} \mapsto y$ between them. Among all possible functions f, the class of *linear functions* $f_\mathbf{u}(\mathbf{x}) = \sum_{j=1}^n x_j u_j = \mathbf{x}^\top \mathbf{u}$ with weight vector $\mathbf{u}$ sticks out: they are simple to handle, very intuitive and enjoy many favourable analytical and algorithmic properties.

In the following chapter, we will first introduce some concepts of statistical inference in a general setting and apply them to the modelling of dependencies $\mathbf{x} \mapsto y$. We will then introduce and discuss estimation, inference and experimental design in linear models with Gaussian noise. Further, we will look at two generalisations thereof: the *generalised linear model* (GLM), where the likelihood can be non-Gaussian, and the *Gaussian process* (GP) model, a kernelised variant, where the functional dependency is linear in a different space and thus non-linear in the covariates $\mathbf{X}$.

Generalised linear models are cornerstones of applied statistics and machine learning. The domains of application range from computer vision, bioinformatics over adaptive filtering and control to neuroscience as well as information retrieval.

The goal of the chapter is to set up a consistent notation and to deliver a high-level overview of the connections between the probabilistic models and inference techniques used in this thesis, especially the theoretical chapter 3. All following application chapters 4, 5 and 6 contain back references but can nevertheless be read on their own. Also, the chapter contrasts frequentist and Bayesian techniques to provide a better link to the statistics literature.

2.1 Statistical inference and decision theory

Statistical inference in its most general form is the process of drawing conclusions from a probabilistic model given a finite sample – the dataset $\mathcal{D}$. Another term expressing the same thing is *induction* or *learning from examples*, where general rules are obtained from a few representative observations. Probabilistic models are supposed to mimic aspects of noisy physical processes in the real world. We denote them formally by a family of distributions $\mathbb{P}_\rho(\mathcal{D})$ over the dataset $\mathcal{D}$ with unknown parameter $\boldsymbol{\rho}$. The resulting conclusions are intended to either yield a prediction of what is going to happen in the future, what could have happened in the past or to lead to a specific decision suggesting an interaction with the world. Probability theory is the natural way to represent noise in the data acquisition process or incomplete knowledge of the underlying process itself.

We will focus on decision making in the following since conclusions of any kind drawn from the data can be seen as a decision; decision theory allows a unified treatment of point estimation, interval estimation and hypothesis testing. A decision is modelled by a decision function $\delta : \mathcal{D} \mapsto \hat{\boldsymbol{\rho}}$ that – based on the data $\mathcal{D}$ – outputs a specific choice $\hat{\boldsymbol{\rho}}$ for the unknown parameter $\boldsymbol{\rho}$ of the model. The quality of a specific decision is formalised by a loss function $\ell(\hat{\boldsymbol{\rho}}, \boldsymbol{\rho}) \in \mathbb{R}$ that measures how much it costs if we use $\hat{\boldsymbol{\rho}}$ given that the actual value is $\boldsymbol{\rho}$. It is a measure of discrepancy between the decision $\hat{\boldsymbol{\rho}} = \delta(\mathcal{D})$ and the parameter $\boldsymbol{\rho}$.

Treating the probabilistic model $\mathbb{P}_{\rho}(\mathcal{D})$ as fixed for now, the *risk* of using the decision rule δ

$$R(\delta, \mathcal{D}, \boldsymbol{\rho}) = \ell(\delta(\mathcal{D}), \boldsymbol{\rho}) \tag{2.1}$$

depends on two quantities: the data $\mathcal{D}$ and the parameter $\boldsymbol{\rho}$. There are two complementary approaches to designing decision functions δ differing in the respective probabilistic interpretation of $\mathcal{D}$ and $\boldsymbol{\rho}$: the Bayesian and the frequentist or Fisherian perspectives. Both schools have their relative merits and shortcomings and many practical problem settings such as experimental design can benefit from the interplay of both [Bayarri and Berger, 2004].

The following exposition is based on an inspiring lecture [Jordan, 2009] and a comprehensive book [Schervish, 1995, ch 3].

2.1.1 Frequentist decision theory

At the core of the frequentist approach is the interpretation of the dataset as being a sample of a random variable. Therefore the *frequentist risk* or *generalisation error*

$$R_F(\delta, \boldsymbol{\rho}) = \mathbb{E}_{\mathbb{P}_{\rho}(\mathcal{D})}\left[\ell(\delta(\mathcal{D}), \boldsymbol{\rho})\right] \tag{2.2}$$

is defined as the expected risk (equation 2.1) over the dataset. This eliminates the dependency on $\mathcal{D}$ based on the idea that our specific dataset is only one possible realisation; we could have gotten different ones. Unfortunately, the expectation cannot be done analytically in most interesting cases.

Theoretically, there are at least two strategies to select an optimal decision function $\delta^{\star}$. The *minimax estimator*

$$\delta^{\star} = \arg\min_{\delta} \max_{\rho} R_F(\delta, \boldsymbol{\rho})$$

is the most pessimistic estimate. It chooses the decision function in light of the most adversarial parameter that exists. While offering clear worst-case guarantees, a minimax estimate can turn out to be overly pessimistic in practice, where the average case scenario is captured by the minimal Bayes risk estimator or *Bayes estimator*

$$\delta^{\star} = \arg\min_{\delta} \mathbb{E}_{\mathbb{P}(\rho)}\left[R_F(\boldsymbol{\rho}, \delta)\right] = \arg\min_{\delta} R_A(\delta).$$

The average risk or Bayes risk $R_A(\delta)$ is the expected risk under a *prior distribution* $\mathbb{P}(\boldsymbol{\rho})$ over the parameters qualifying therefore as a hybrid method between the Bayesian and frequentist points of view.

In general, frequentist methods are designed to give trustable answers if used repeatedly. For example in software engineering, where many users run a system on many different inputs, minimax parameter estimates are appropriate.

Structural and empirical risk minimisation

Since the expectation $\mathbb{E}_{\mathbb{P}_{\rho}(\mathcal{D})}\left[\ell(\delta(\mathcal{D}), \boldsymbol{\rho})\right]$ over the dataset in the generalisation error $R_F(\delta, \boldsymbol{\rho})$ is most likely intractable, one has to resort to approximation or bounding techniques.

One approach derives upper bounds on the risk $B_F(\delta, \boldsymbol{\rho}) \geq R_F(\delta, \boldsymbol{\rho})$ and uses them as building blocks to shape the objective characterising the estimator. Known under the name of *structural risk minimisation* (SRM) [Vapnik, 1998], it is a successful principle for overfitting prevention in linear classification models, where the SRM term is a function of the margin of the separating hyperplane. SRM implements the principle of complexity control by limiting the capacity of the predictor.

The upper bound of the SRM approach alone is not sufficient to train a predictor since it does not depend on the data $\mathcal{D}$. By replacing the expectation $\mathbb{E}_{\mathbb{P}_{\rho}(\mathcal{D})}\left[\ell(\delta(\mathcal{D}), \boldsymbol{\rho})\right]$ with an empirical sum over the particular dataset $\mathcal{D}$, one gets an estimate for the generalisation error, the so-called *empirical risk* $\hat{R}_F(\delta, \boldsymbol{\rho})$ giving rise to the principle of *empirical risk minimisation* (ERM). Better estimates can be obtained by resampling techniques such as bootstrapping, leave-one-out estimators or cross-validation (CV) [Wasserman, 2005].

Binary classification

In *support vector machines* (SVMs) [Schölkopf and Smola, 2002], both SRM and ERM are used. There are also approaches to include the minimax principle [Davenport et al., 2010]. Here, $\delta_{\mathbf{u}}(\mathbf{x}) = \text{sign}(\mathbf{u}^\top \mathbf{x})$ is a linear classifier parametrised by the weights $\mathbf{u}$ whose quality is measured by the *hinge loss* $\ell(\mathbf{x}, y, \mathbf{u}) = \max(0, -y \cdot \mathbf{u}^\top \mathbf{x})$. The empirical risk, a simple sum over the dataset $\hat{R}_F(\mathbf{u}) = \sum_{i=1}^m \ell(\mathbf{x}_i, y_i, \mathbf{u})$, is combined with the complexity penalty $\mathbf{u}^\top \mathbf{u}$ into the *regularised risk* $\mathbf{u}^\top \mathbf{u} + C \cdot \hat{R}_F(\mathbf{u})$, where C balances the relative contributions. The parameter C is typically set by minimising a CV estimate of $R_F(\delta_{\mathbf{u}}, C)$.

2.1.2 Bayesian perspective

Also starting from the risk of equation 2.1, the Bayesian method computes an average over parameters rather than over the data

$$R_B(\mathcal{D}, \delta) = \mathbb{E}_{\mathbb{P}(\boldsymbol{\rho}|\mathcal{D})}\left[\ell(\delta(\mathcal{D}), \boldsymbol{\rho})\right]. \tag{2.3}$$

The expectation is taken w.r.t. the posterior distribution $\mathbb{P}(\boldsymbol{\rho}|\mathcal{D})$ that is obtained by applying Bayes' rule

$$\mathbb{P}(\boldsymbol{\rho}|\mathcal{D}) = \frac{\mathbb{P}(\boldsymbol{\rho})\mathbb{P}(\mathcal{D}|\boldsymbol{\rho})}{\int \mathbb{P}(\boldsymbol{\rho})\mathbb{P}(\mathcal{D}|\boldsymbol{\rho})\mathrm{d}\boldsymbol{\rho}} = \frac{\mathbb{P}(\boldsymbol{\rho})\mathbb{P}(\mathcal{D}|\boldsymbol{\rho})}{\mathbb{P}(\mathcal{D})} \tag{2.4}$$

that follows from the definition of conditional probability. Here, the prior $\mathbb{P}(\boldsymbol{\rho})$ describes the initial belief about the parameter $\boldsymbol{\rho}$, the *posterior* $\mathbb{P}(\boldsymbol{\rho}|\mathcal{D})$ contains the uncertainty about $\boldsymbol{\rho}$ after seeing the data $\mathcal{D}$ and the *likelihood* of the parameters or sampling distribution $\mathbb{P}(\mathcal{D}|\boldsymbol{\rho})$ can generate synthetic data given a fixed parameter $\boldsymbol{\rho}$. The normaliser $\mathbb{P}(\mathcal{D})$ is termed the *marginal likelihood* or *evidence* and is used to compare models (see section on marginal likelihood II and Bishop [2006], MacKay [2005]).

Optimal decisions using Bayes estimators are obtained by minimising the risk of equation 2.3

$$\delta^\star = \arg\min_\delta R_B(\mathcal{D}, \delta).$$

For some loss functions ℓ, the Bayes estimators can be computed exactly and correspond to specific properties of the posterior $\mathbb{P}(\boldsymbol{\rho}|\mathcal{D})$ as listed in the following table.

loss function $\ell(\hat{\boldsymbol{\rho}}, \boldsymbol{\rho})$	$\|\hat{\boldsymbol{\rho}} - \boldsymbol{\rho}\|_2$	$\|\hat{\boldsymbol{\rho}} - \boldsymbol{\rho}\|_1$	$\|\hat{\boldsymbol{\rho}} - \boldsymbol{\rho}\|_0$
Bayes estimator $\hat{\boldsymbol{\rho}} = \delta^\star(\mathcal{D})$	mean	centroid (multivariate median)	mode

Table 2.1: Loss functions and Bayes estimators

The maximum a posteriori (MAP) estimator selecting the posterior mode is simple to compute in practice, but – as any Bayesian estimator – it has two inconvenient properties: first, the loss function is questionable since it penalises all parameters except for the correct $\boldsymbol{\rho}$ by the same amount. Second, it is not invariant under a reparametrisation $\boldsymbol{\xi} : \boldsymbol{\rho} \mapsto \boldsymbol{\xi}(\boldsymbol{\rho})$ (continuous bijection) since in general, we have

$$\boldsymbol{\xi}\left(\arg\min_{\boldsymbol{\rho}} \mathbb{P}(\boldsymbol{\rho}|\mathcal{D})\right) \neq \arg\min_{\boldsymbol{\xi}} \mathbb{P}(\boldsymbol{\xi}(\boldsymbol{\rho})|\mathcal{D}) \left|\det\left(\frac{\partial \boldsymbol{\xi}(\boldsymbol{\rho})}{\partial \boldsymbol{\rho}^\top}\right)\right|$$

implying that we can move around the mode as much as we want by changing the Jacobi correction term $|\det(\frac{\partial \boldsymbol{\xi}(\boldsymbol{\rho})}{\partial \boldsymbol{\rho}^\top})|$. Equality holds for linear transformations $\boldsymbol{\xi}$. Bayesian estimators are only invariant under reparametrisation if the loss is transformed as well (see appendix D.3).

Binary classification

In the example of binary pattern classification, where a class $y_*^\star \in \{0, 1\}$ has to be assigned to a pattern $\mathbf{x}_*$, the Hamming loss $\ell(\hat{y}, y) = \hat{y} \cdot (1 - y) + (1 - \hat{y}) \cdot y$ is appropriate if there is no prior information on the class labels available. From the posterior $\mathbb{P}(\boldsymbol{\rho}|\mathcal{D})$, we can compute the predictive distribution

$$\mathbb{P}(y_*|\mathcal{D}) = \int \mathbb{P}(y_*|\boldsymbol{\rho})\mathbb{P}(\boldsymbol{\rho}|\mathcal{D})\mathrm{d}\boldsymbol{\rho}$$

and take the Bayesian expectation of the loss function

$$R_B(\mathcal{D}, \hat{y}_*) = \int \mathbb{P}(y_*|\mathcal{D})\ell(\hat{y}_*, y_*)\mathrm{d}y_* = \mathbb{P}(1 - \hat{y}_*|\mathcal{D}) = 1 - \mathbb{P}(\hat{y}_*|\mathcal{D}).$$

The optimal decision rule is hence given by

$$y_*^\star = \arg\min_{\hat{y}_*} R_B(\mathcal{D}, \hat{y}_*) = \arg\max_{\hat{y}_*} \mathbb{P}(\hat{y}_*|\mathcal{D}) = \frac{1}{2} + \frac{1}{2}\mathrm{sign}\left(\mathbb{P}(\hat{y}_* = 1|\mathcal{D}) - \frac{1}{2}\right)$$

that is, we have to choose the most probable class $y_*^\star$ in order to obtain the optimal decision. Here, $\mathrm{sign}(x) \in \{\pm 1\}$ computes the sign of x, where 0 is mapped to $+1$.

Maximum likelihood II and hyperparameters

Sometimes, it is useful to treat some parameters $\boldsymbol{\theta} \subset \boldsymbol{\rho}$ in a slightly different way by interpreting them as *hyperparameters*. A hyperparameter, in loose terms, is a parameter at a higher level in a hierarchical model such as the weight C between the terms in SVM models (section 2.1.1) or a parameter for which correct marginalisation is very hard.

The *maximum likelihood II* approach, sometimes called marginal likelihood or evidence maximisation proceeds by computing the posterior of the hyperparameters

$$\mathbb{P}(\boldsymbol{\theta}|\mathcal{D}) \propto \mathbb{P}(\mathcal{D}|\boldsymbol{\theta})\mathbb{P}(\boldsymbol{\theta}),$$

where $\mathbb{P}(\mathcal{D}|\boldsymbol{\theta})$ is the marginal likelihood for a fixed hyperparameter value $\boldsymbol{\theta}$. Using MAP estimation, the mode

$$\boldsymbol{\theta}^\star = \arg\max_{\boldsymbol{\theta}} \mathbb{P}(\boldsymbol{\theta}|\mathcal{D})$$

is computed and used subsequently instead of $\mathbb{P}(\boldsymbol{\theta}|\mathcal{D})$. All criticism made to MAP estimation applies to that approach but also all asymptotic virtues of maximum likelihood are present, making this *empirical Bayes* strategy always a pragmatic decision in light of computational complexity or analytical intractability.

Although conceptually very elegant and clear, the Bayesian viewpoint has been criticised for its subjectivity introduced by the choice of prior. However, if a domain expert provides valuable prior knowledge, why shouldn't we use it? The Bayesian approach is most suited to applications, where repetition is not the major concern but an assessment of a concrete task on a concrete dataset is at the focus of interest. Its elegance is due to the fact that every quantity in the model is treated as a random variable. Modelling corresponds to making explicit the statistical dependencies between the random variables. A prediction is done by computing the marginal distribution w.r.t. the variable of interest and decision making corresponds to selecting the point estimate minimising the expected loss under the predictive distribution. Even though these guidelines are very clear in theory, in practice most of the integrals are intractable; therefore most of the work goes into approximate numerical integration methods as detailed in section 2.5.

2.2 The Gaussian linear model

The Gaussian linear model for linear dependencies $\mathbf{x} \mapsto y$ is a very interesting special case of a *parametric model*, where both inference and estimation are analytically tractable and closely related to each other.

Assuming independence between individual measurements y_i and normally distributed additive errors $\varepsilon_i \sim \mathcal{N}(0, \sigma^2)$ we get the linear relation

$$y_i = f_{\mathbf{u}}(\mathbf{x}_i) + \varepsilon_i, \quad i = 1..m, \quad \mathbf{y} = \mathbf{X}\mathbf{u} + \boldsymbol{\varepsilon} \tag{2.5}$$

between the covariates $\mathbf{X}$ and the observations $\mathbf{y}$ summarised by the likelihood function

$$\mathbb{P}(\mathbf{y}|\mathbf{u}) \;\; = \;\; \prod_{i=1}^{m} \mathbb{P}(y_i|\mathbf{x}_i^\top \mathbf{u}) = \mathcal{N}(\mathbf{y}|\mathbf{X}\mathbf{u}, \sigma^2 \mathbf{I}).$$

2.2.1 Frequentist estimation

In case we want to come up with a single representative $\hat{\mathbf{u}}$ for the unknown weight $\mathbf{u}$ to be used in subsequent predictions, a common approach is to consider the popular *maximum likelihood* (ML) estimator

$$\hat{\mathbf{u}}_{\text{ML}} = \arg\max_{\mathbf{u}} \mathbb{P}(\mathbf{y}|\mathbf{u}) = \arg\min_{\mathbf{u}} \left[-\ln \mathbb{P}(\mathbf{y}|\mathbf{u}) \right],$$

where $-\ln \mathbb{P}(\mathbf{y}|\mathbf{u})$ is called the *data fit term*. Informally, the ML estimator can be interpreted as a MAP estimator under a flat prior. Besides several invariance properties (see appendix D.2), the ML estimator has a lot of asymptotic properties: it is asymptotically *unbiased*[1], *efficient*[2] and *normal*[3].

For Gaussian likelihood, the ML estimator is also called the *ordinary least squares* (OLS) estimator

$$\hat{\mathbf{u}}_{\text{OLS}} = \arg\max_{\mathbf{u}} \mathcal{N}(\mathbf{y}|\mathbf{X}\mathbf{u}, \sigma^2 \mathbf{I}) = \arg\min_{\mathbf{u}} ||\mathbf{X}\mathbf{u} - \mathbf{y}||^2 \Leftrightarrow \mathbf{X}^\top \mathbf{X} \hat{\mathbf{u}}_{\text{OLS}} = \mathbf{X}^\top \mathbf{y} \tag{2.6}$$

minimising the squared distance between predictions and measurements. The estimator $\hat{\mathbf{u}}_{\text{OLS}}$ is a random variable with mean $\mathbb{E}\left[\hat{\mathbf{u}}_{\text{OLS}}\right] = \mathbf{u}$, covariance matrix $\mathbb{V}\left[\hat{\mathbf{u}}_{\text{OLS}}\right] = \sigma^2 (\mathbf{X}^\top \mathbf{X})^{-1}$ and Gaussian distribution. Note that the unknown $\mathbf{u}$ is regarded as a deterministic quantity. If the normal equations (equation 2.6) are underdetermined or badly conditioned, *regularised* or *penalised least squares* (PLS) estimators

$$\hat{\mathbf{u}}_{\text{PLS}} = \arg\min_{\mathbf{u}} \gamma^{-1} \, ||\mathbf{B}\mathbf{u}||_p^p + ||\mathbf{X}\mathbf{u} - \mathbf{y}||^2 \tag{2.7}$$

can be used, where $||\mathbf{B}\mathbf{u}||_p^p$ is called the *regulariser*, and where the matrix $\mathbf{B} \in \mathbb{R}^{q \times n}$ encodes the domain of penalisation. Via $\mathbf{B} = \mathbf{I}$, we directly penalise large values of $\mathbf{u}$, and by setting $\mathbf{B}$ to the finite difference matrix, we can penalise high deviations between neighbouring components of $\mathbf{u}$. As a result, the absolute values of the components of $\hat{\mathbf{u}}_{\text{PLS}}$ will be smaller than the absolute value of $\hat{\mathbf{u}}_{\text{OLS}}$ since the penaliser $||\mathbf{B}\mathbf{u}||_p^p$ will shift the optimal value towards $\mathbf{0}$. This behaviour is typically denoted by the term *shrinkage* [Stein, 1956, Copas, 1983]. In LS-estimation, shrinkage does not depend on the measurements $\mathbf{y}$ and is therefore *non-adaptive* or *non-selective*. Shrinkage estimators are an active research topic in statistics. Especially, $p = 1$ [Tibshirani, 1996, Breiman, 1995] recently attracted a lot of attention as the *LASSO* (least absolute shrinkage and selection operator) because the resulting estimators are *sparse* with many entries being zero. For $\mathbf{B} = \mathbf{I}$ and $p = 2$, the technique is known as *ridge regression* in statistics [e.g. Hastie et al., 2009] or *Tikhonov regularisation* [Tikhonov and Arsenin, 1977] in the inverse problems literature.

[1] $\lim_{m\to\infty} \mathbb{E}[\hat{\mathbf{u}}_{\text{ML}}] - \mathbf{u} = \mathbf{0}$

[2] $\lim_{m\to\infty} \mathbb{V}[\hat{\mathbf{u}}_{\text{ML}}] - \mathbf{V} = \mathbf{0}$, where $\mathbf{V}$ is the variance from the Cramér-Rao lower bound of section 2.6.1.

[3] $\hat{\mathbf{u}}_{\text{ML}} \overset{m\to\infty}{\sim} \mathcal{N}(\mathbb{E}[\hat{\mathbf{u}}_{\text{ML}}], \mathbb{V}[\hat{\mathbf{u}}_{\text{ML}}])$

2.2.2 Bayesian inference

By combining a prior distribution $\mathbb{P}(\mathbf{u})$ over the unknown weights $\mathbf{u}$ with the likelihood $\mathbb{P}(\mathbf{y}|\mathbf{u})$, we obtain the posterior distribution $\mathbb{P}(\mathbf{u}|\mathbf{y}) \propto \mathbb{P}(\mathbf{u})\mathbb{P}(\mathbf{y}|\mathbf{u})$, which represents the remaining uncertainty about the unknown and therefore random weights $\mathbf{u}$ in Bayesian inference. Assuming a Gaussian prior $\mathbb{P}(\mathbf{u}) \propto \prod_{i=1}^{q} \mathcal{N}(s_i|0, \sigma^2\gamma_i)$, where $\mathbf{s} = \mathbf{B}\mathbf{u}$, the posterior is of the form

$$\mathbb{P}(\mathbf{u}|\mathbf{y}) = \mathcal{N}\left(\mathbf{u}|\mathbf{A}^{-1}\mathbf{X}^\top\mathbf{y}, \sigma^2\mathbf{A}^{-1}\right), \, \mathbf{\Gamma} = \mathrm{dg}(\boldsymbol{\gamma}), \; \mathbf{A} = \mathbf{X}^\top\mathbf{X} + \mathbf{B}^\top\mathbf{\Gamma}^{-1}\mathbf{B}. \tag{2.8}$$

Therefore, the outcome of a Bayesian procedure is the posterior distribution over $\mathbf{u}$ in contrast to a single estimate $\hat{\mathbf{u}}$. Note that for full rank $\mathbf{X}^\top\mathbf{X}$, the OLS and the PLS estimators correspond to maxima of posteriors (MAP) with prior variances $\sigma^2\gamma_i$ being all equal $\boldsymbol{\gamma} = \gamma\mathbf{1}$, which holds for many other estimators, as well.

$$\begin{aligned} \hat{\mathbf{u}}_{\text{PLS}} &= \arg\max_{\mathbf{u}} \mathbb{P}(\mathbf{u}|\mathbf{y}), \quad p = 2 \\ \hat{\mathbf{u}}_{\text{OLS}} &= \arg\max_{\mathbf{u}} \mathbb{P}(\mathbf{u}|\mathbf{y}), \quad \gamma \to \infty \end{aligned}$$

In the linear Gaussian case, mean, mode and centroid are the same, which means that the $\hat{\mathbf{u}}_{\text{OLS}}$ and $\hat{\mathbf{u}}_{\text{PLS}}$ for $p = 2$ all coincide with the Bayesian estimator under a wide range of loss functions. When it comes to variance/covariance estimation and to experimental design based thereupon, however, there are quite severe differences (see section 2.6.6).

2.3 The generalised linear model

Often, the observations $\mathbf{y}$ cannot be described by linear functions of the covariates $\mathbf{X}$ directly. For example, in *binary classification*, the class probabilities are numbers between 0 and 1. Count data is strictly positive imposing non-negativity constraints on $\mathbf{y}$. In both cases, $\mathbf{y}$ cannot be modelled as a noisy version of $\mathbf{X}\mathbf{u}$. A *generalised linear model* (GLM) [Nelder and Wedderburn, 1972] assumes that an observation y follows an exponential family distribution whose mean is a nonlinear function of $\mathbf{x}^\top\mathbf{u}$. In other words, the likelihood $\mathbb{P}(y|\mathbf{u})$ can be written as $\mathbb{P}(y|\mathbf{x}^\top\mathbf{u})$. A concise treatment is given in McCullagh and Nelder [1989]; logistic regression is discussed by Hastie et al. [2009, §4.4]. Formally, a GLM consists of a *linear predictor* $\boldsymbol{\eta} = \mathbf{X}\mathbf{u}$ and a pointwise *link function* $g : \mu \mapsto \eta$ relating the linear predictor to the expectation $\mathbb{E}[y] = \mu = g^{-1}(\eta)$. Often, the variance, $\mathbb{V}[y]$ is a simple function of the mean μ. Table 2.2 lists three common choices of link functions along with their inducing likelihood.

Exponential family distribution	Normal y	Poisson y	Binomial y
Name of the link function g	identity	log	logistic
Name of the GLM	regression	Poisson regression	logistic regression
mean $\mathbb{E}[y] = \mu = g^{-1}(\eta)$	$\mu = \eta \in \mathbb{R}$	$\mu = e^\eta \in \mathbb{R}_+$	$\mu = \frac{1}{1+e^{-\eta}} \in [0, 1]$
variance $\mathbb{V}[y] = v(\mu)$	σ^2	μ	$\mu(1-\mu)$
likelihood $\mathbb{P}(y|\mathbf{u}) = \mathbb{P}(y|\mathbf{x}^\top\mathbf{u})$	$\mathcal{N}(y|\mathbf{x}^\top\mathbf{u}, \sigma^2)$	$\frac{\mu^y}{y!}e^{-\mu}$, $\mu = \exp(\mathbf{x}^\top\mathbf{u})$	$\left(1+\exp(-y\cdot\mathbf{x}^\top\mathbf{u})\right)^{-1}$

Table 2.2: Common link functions in the generalised linear model

With these definitions in place, one can – for a fixed parameter $\mathbf{u}$ and say logistic link – predict y_* from $\mathbf{x}_*$ via

$$\mathbb{E}[y_*] = \frac{1}{1+\exp(-\mathbf{x}_*^\top\mathbf{u})}, \; \mathbb{V}[y_*] = \mathbb{E}[y_*]\left(1-\mathbb{E}[y_*]\right).$$

2.3.1 Frequentist estimation

Model fitting is done using the ML estimator

$$\hat{\mathbf{u}}_{\mathrm{ML}} = \arg\min_{\mathbf{u}} \left[-\sum_{i=1}^{m} \ln \mathbb{P}(y_i|\mathbf{x}_i^\top \mathbf{u}) \right] = \arg\min_{\mathbf{u}} \ell(\mathbf{X}\mathbf{u}).$$

One approach for the optimisation of $\ell(\mathbf{X}\mathbf{u})$ w.r.t. $\mathbf{u}$ is the Newton-Raphson algorithm, where a local quadratic approximation to ℓ is minimised in every iteration step. The Newton descent direction $\mathbf{d}$ is computed from the gradient vector $\mathbf{g}$ and the Hessian matrix $\mathbf{H}$ by $\mathbf{d} = -\mathbf{H}^{-1}\mathbf{g}$. Defining the negative log-likelihood vector $\boldsymbol{\ell}$ with $\ell_i = \ell_i(\mathbf{x}_i^\top \mathbf{u}) = -\ln \mathbb{P}(y_i|\mathbf{x}_i^\top \mathbf{u})$ as well as its first two derivatives $\boldsymbol{\ell}' = \left[\ell_i'(\mathbf{x}_i^\top \mathbf{u})\right]_i$ and $\mathbf{L}'' = \left[\ell_i''(\mathbf{x}_i^\top \mathbf{u})\right]_{ii}$, we obtain

$$\mathbf{g} = \frac{\partial \ell(\mathbf{X}\mathbf{u})}{\partial \mathbf{u}} = \mathbf{X}^\top \boldsymbol{\ell}' \quad \text{and} \quad \mathbf{H} = \frac{\partial^2 \ell(\mathbf{X}\mathbf{u})}{\partial \mathbf{u} \partial \mathbf{u}^\top} = \mathbf{X}^\top \mathbf{L}'' \mathbf{X}$$

leading to the linear system

$$\mathbf{d} = -\mathbf{H}^{-1}\mathbf{g} \Leftrightarrow \mathbf{X}^\top \mathbf{L}'' \mathbf{X} \mathbf{d} = -\mathbf{X}^\top \boldsymbol{\ell}',$$

which is an $\mathbf{L}''$-reweighted variant of the LS problem in equation 2.6, where $-\boldsymbol{\ell}'$ takes the role of $\mathbf{y}$. Therefore the Newton-Raphson algorithm to find the ML estimator in GLMs is called *iteratively reweighted least squares* (IRLS) [Green, 1984] .

2.3.2 Bayesian inference

As in the Gaussian linear model, Bayesian inference starts with a prior $\mathbb{P}(\mathbf{u})$. The likelihood function $\mathbb{P}(\mathbf{y}|\mathbf{u})$ is no longer restricted to be Gaussian rendering the posterior $\mathbb{P}(\mathbf{u}|\mathbf{y})$ analytically intractable in most cases. Therefore, approximate inference techniques enter the stage. We will discuss these in section 2.5; for a good overview see Bishop [2006, Ch. 10].

2.4 The Gaussian process model

In many applications, the functional relationship f between data points $\mathbf{x}$ and observations y is non-linear even though the noise might still be Gaussian. Gaussian process (GP) models are a powerful *nonparametric* way to make inference over nonlinear functions f. They were used in geostatistics under the name *kriging* [Matheron, 1973], applied to spatial statistics [Ripley, 1981] and brought as a high-dimensional regression tool into machine learning [Williams and Rasmussen, 1996] with proper probabilistic interpretations. We will informally motivate them as linear models in high-dimensional *feature spaces* and show that the computations have the same structure as in the linear case.

Explicit feature expansion

One approach to transfer linear technology to non-linear models proceeds by defining explicit *basis* or *feature functions* $\psi_1(\mathbf{x}), .., \psi_d(\mathbf{x})$ and assuming the function to be linear in $\psi_j(\mathbf{x})$ instead of x_i itself

$$y_i = f(\mathbf{x}_i) + \varepsilon_i = \sum_{j=1}^{d} u_j \psi_j(\mathbf{x}_i) + \varepsilon_i = \mathbf{u}^\top \boldsymbol{\psi}(\mathbf{x}_i) + \varepsilon_i.$$

Estimation, inference and design are exactly the same as in the linear Gaussian case, only the data matrix $\mathbf{X}$ has to be replaced by the feature matrix $\boldsymbol{\Psi} = [\psi_j(\mathbf{x}_i)]_{ij}$ in all computations. However, if the number of feature functions d becomes large[4], ML estimation cannot be successful due to the big number of parameters. One has to resort to regularised estimators or Bayesian inference.

[4] We could choose all polynomials up to degree 3 leading to $d = n^3$, where n is the dimension of a data point $\mathbf{x}_i$.

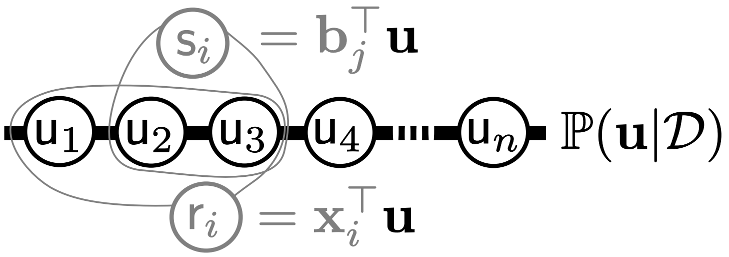

$s_i = \mathbf{b}_j^\top \mathbf{u}$
u_1
u_2
u_3
u_4
u_n
$\mathbb{P}(\mathbf{u}|\mathcal{D})$
$r_i = \mathbf{x}_i^\top \mathbf{u}$

Figure 2.1: Graphical model of the general posterior
Graphical model of the general posterior $\mathbb{P}(\mathbf{u}|\mathcal{D})$ *as a factor graph of Gaussian potentials on* r_i *and non-Gaussian potentials on* s_j. *The variables* $\mathbf{u}$ *are densely coupled. Distribution models of this sort are called* undirected graphical models *or* Markov random fields *[Lauritzen, 1996].*

Implicit feature functions and the function space view

A dual approach using implicit feature functions is known as the function space view on GPs [Rasmussen and Williams, 2006, Seeger, 2004]. Starting from a Gaussian prior on the weights $\mathbb{P}(\mathbf{u}) = \mathcal{N}(\mathbf{u}|\mathbf{0}, \mathbf{I})$ of the basis expansion for $f(\mathbf{x})$ in terms of the feature functions $\psi_i(\mathbf{x})$, we can compute the mean and covariance of the Gaussian distribution over $\mathbf{f} = [f(\mathbf{x}_1), .., f(\mathbf{x}_m)]^\top = \mathbf{\Psi u}$ as

$$\mathbb{E}\left[\mathbf{f}\right] = \mathbf{\Psi}\mathbb{E}\left[\mathbf{u}\right] = 0 \quad \text{and} \quad \mathbb{V}[\mathbf{f}] = \mathbb{E}\left[\mathbf{f}\mathbf{f}^\top\right] = \mathbf{\Psi}\mathbb{E}\left[\mathbf{u}\mathbf{u}^\top\right]\mathbf{\Psi}^\top = \mathbf{\Psi}\mathbf{\Psi}^\top =: \mathbf{K}.$$

Hence, we can write $\mathbb{P}(f|\mathbf{X}) =: \mathbb{P}(\mathbf{f}) = \mathcal{N}(\mathbf{f}|\mathbf{0}, \mathbf{K})$ to emphasise that we deal with a distribution over functions f specified at the locations $\mathbf{x}_i$. Here, the matrix $\mathbf{K}$ contains the covariances $K_{ij} = k(\mathbf{x}_i, \mathbf{x}_j) = [\boldsymbol{\psi}(\mathbf{x}_i)]^\top \boldsymbol{\psi}(\mathbf{x}_j)$. We say that the function $f(\cdot)$ follows a GP prior distribution with covariance function $k(\cdot, \cdot)$ and mean function $m(\mathbf{x}) = 0$: $f(\cdot) \sim \mathcal{GP}\left(m(\cdot), k(\cdot, \cdot)\right)$. This enables us to do Bayesian inference over the latent function values $\mathbf{f} = [f_1, .., f_m]^\top$ instead of the weights $\mathbf{u} = [u_1, .., u_d]^\top$. We do not have to compute a single evaluation of $\boldsymbol{\psi}(\mathbf{x}_i)$ explicitly; the feature functions enter only implicitly through the positive definite covariance function $k(\cdot, \cdot)$. This property became popular under the name *kernel trick*. Therefore, the dimension of the feature space d becomes computationally irrelevant since the complexity scales with m^3 rather than d^3. GPs are a member of the family of *kernel machines* [Schölkopf and Smola, 2002] – kernel being only a synonym for covariance function.

Gaussian process regression and linear regression

To see the strong formal similarities with linear Gaussian regression, we consider a GP model with Gaussian likelihood $\mathbb{P}(y_i|f_i) = \mathcal{N}(y_i|f_i, \sigma_n^2)$. The posterior distribution is given by

$$\begin{aligned}\mathbb{P}(\mathbf{f}|\mathbf{y}) &\propto \mathcal{N}(\mathbf{f}|\mathbf{0}, \mathbf{K})\mathcal{N}(\mathbf{y}|\mathbf{f}, \sigma_n^2\mathbf{I}) \\ &= \mathcal{N}(\mathbf{f}|\sigma_n^{-2}\mathbf{A}^{-1}\mathbf{y}, \mathbf{A}^{-1}),\ \mathbf{A} = \mathbf{K}^{-1} + \sigma_n^{-2}\mathbf{I},\end{aligned}$$

which can be recognised as an instance of equation 2.8 with $\sigma = 1$, $\mathbf{B} = \mathbf{I}$, $\gamma = \sigma_n^2$ and the formal replacements $\mathbf{X}^\top\mathbf{y} \leftarrow \mathbf{y}$, $\mathbf{X}^\top\mathbf{X} \leftarrow \mathbf{K}^{-1}$.

In case of non-Gaussian likelihood functions for classification or robust regression, the posterior cannot be computed in closed form as in the linear Gaussian case, but as it can be seen in the next section, we have a wide range of approximate inference techniques available that also apply to the nonlinear case.

2.5 Approximate Bayesian inference

In the following, we will look at GLMs with Gaussian and non-Gaussian contributions. We will develop a unifying notation and introduce the most prominent methods allowing us to compute an approximation to the Bayesian posterior.

2.5.1 Modelling framework

We start from two observations: first, a GLM (see section 2.3) can have different link functions for different components of the linear predictor $\boldsymbol{\eta}$. For example y_3 could be Gaussian but y_{11} could be Poisson. Second, the prior needed for Bayesian inference can formally be treated in the same way as the likelihood. For example, we can rewrite a general Gaussian prior $\mathbb{P}(\mathbf{u}) = \mathcal{N}(\mathbf{u}|\boldsymbol{\mu}, \boldsymbol{\Sigma})$ as a product of one-dimensional distributions acting on linear projections of the unknown variable $\mathbf{u}$

$$\begin{aligned}\mathbb{P}(\mathbf{u}) &= \mathcal{N}(\mathbf{u}|\boldsymbol{\mu}, \boldsymbol{\Sigma}) = \mathcal{N}\left(\mathbf{V}^\top\boldsymbol{\mu}|\mathbf{V}^\top\mathbf{u}, \frac{\sigma^2}{\sigma^2}\boldsymbol{\Lambda}\right) = \overbrace{\sigma^n\,|\boldsymbol{\Sigma}|^{-\frac{1}{2}}}^{C_\mathcal{N}}\mathcal{N}\left(\sigma\boldsymbol{\Lambda}^{-\frac{1}{2}}\mathbf{V}^\top\boldsymbol{\mu}|\sigma\boldsymbol{\Lambda}^{-\frac{1}{2}}\mathbf{V}^\top\mathbf{u}, \sigma^2\mathbf{I}\right) \\ &:= C_\mathcal{N}\cdot\prod_{i=1}^{n}\mathcal{N}\left(y_j|\mathbf{x}_j^\top\mathbf{u}, \sigma^2\right) = C_\mathcal{N}\cdot\mathcal{N}\left(\mathbf{y}|\mathbf{X}\mathbf{u}, \sigma^2\mathbf{I}\right), \mathbf{X} := \sigma\boldsymbol{\Lambda}^{-\frac{1}{2}}\mathbf{V}^\top, \, \mathbf{y} := \sigma\boldsymbol{\Lambda}^{-\frac{1}{2}}\mathbf{V}^\top\boldsymbol{\mu},\end{aligned}$$

where $\boldsymbol{\Sigma} = \mathbf{V}\boldsymbol{\Lambda}\mathbf{V}^\top$ is the eigenvalue decomposition of the covariance matrix and the factor $C_\mathcal{N}$ is constant in $\mathbf{u}$.

In the following, we will call a 1-dimensional distribution acting on a linear projection of $\mathbf{u}$ a *potential*. This has the advantage, that we can talk about prior and likelihood using the same term. In general, potentials do not need to be normalised; we only assume that the posterior is properly normalisable and decomposes into a product of Gaussian potentials $\mathcal{N}(y_i|r_i, \sigma^2)$, $r_i = \mathbf{x}_i^\top\mathbf{u}$ and non-Gaussian potentials $\mathcal{T}_j(s_j)$, $s_j = \mathbf{b}_j^\top\mathbf{u}$

$$\begin{aligned}\mathbb{P}(\mathbf{u}|\mathcal{D}) &= \frac{1}{Z}C_\mathcal{N}\prod_{i=1}^{m}\mathcal{N}(y_i|\mathbf{x}_i^\top\mathbf{u}, \sigma^2)\cdot C_\mathcal{T}\prod_{j=1}^{q}\mathcal{T}_j(s_j) \\ &\propto \mathcal{N}(\mathbf{y}|\mathbf{X}\mathbf{u}, \sigma^2\mathbf{I})\prod_{j=1}^{q}\mathcal{T}(s_j), \quad Z = \mathbb{P}(\mathcal{D}) = C_\mathcal{N}C_\mathcal{T}\cdot\int\mathcal{N}(\mathbf{y}|\mathbf{X}\mathbf{u}, \sigma^2\mathbf{I})\prod_{j=1}^{q}\mathcal{T}_j(s_j)\mathrm{d}\mathbf{u}.\end{aligned}$$

The factors $C_\mathcal{N}$ and $C_\mathcal{T}$ are normalisation constants needed to evaluate the marginal likelihood $Z = \mathbb{P}(\mathcal{D})$ correctly and they originate from our need to write $\mathbb{P}(\mathbf{u}|\mathcal{D})$ as a product of individual potentials on linear projections of $\mathbf{u}$. Figure 2.1 depicts the decomposition of $\mathbb{P}(\mathbf{u}|\mathbf{y})$ into potentials; note that we have a fully connected model with dense matrices $\mathbf{X}$ and $\mathbf{B}$ so far and figure 2.2 gives an overview of the potentials we use. In classification, the likelihood consists of Bernoulli potentials and the prior contains Gaussian potentials. In sparse classification, the prior would include Laplace potentials leading to a completely non-Gaussian model. They do all fit under the umbrella of posterior distribution given as a product of potentials. Making the GLM perspective more apparent, we can write

$$\begin{bmatrix}\mathbf{r}\\ \mathbf{s}\end{bmatrix} = \begin{bmatrix}\mathbf{X}\\ \mathbf{B}\end{bmatrix}\mathbf{u}, \, \mathbf{y} = \mathbf{r} + \boldsymbol{\varepsilon}, \, \varepsilon_i \sim \mathcal{N}(0, \sigma^2).$$

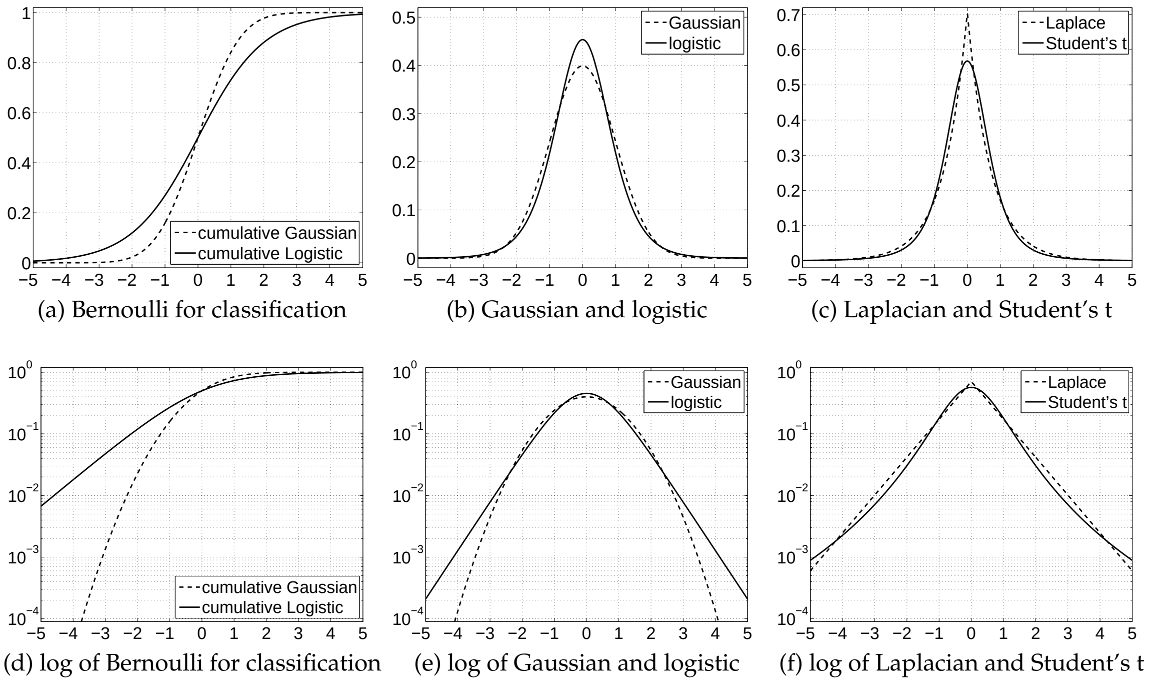
Now that we have fixed the class of models, let us look at some desiderata we have for inference algorithms.

2.5.2 Inference algorithm properties

The first property, we want, is *generality*. We want the inference procedure to not only work for a specific potential but for a large class of them. For example, *super-Gaussian* potentials will play a prominent role. Secondly, we want the inference algorithm to be *scalable*. We want the computational complexity not to increase too strongly if the number of potentials $m + q$ increases. At best we want $\mathcal{O}(m + q)$. The third property is *efficiency* meaning that all available structure in the dependencies represented by $\mathbf{X}$ and $\mathbf{B}$ is used to make the computations as fast as possible. Applications with $\sim 10^5$ potentials require generality, scalability and efficiency. Otherwise estimation, inference and experimental design are impossible.

Classification, class c	Regression	
$\mathcal{T}_{\text{cumGauss}}(s) = \int_{-\infty}^{s} \mathcal{N}(c \cdot t \vert 0,1)\mathrm{d}t$	$\mathcal{T}_{\text{Gauss}}(s)^{\#} = \mathcal{N}(s \vert 0,\sigma^2)$	$\mathcal{T}_{\text{Laplace}}(s)^{\#} = \frac{1}{2b}\exp(-\vert s \vert/b),\ b = \frac{\sigma}{\sqrt{2}}$
$\mathcal{T}_{\text{cumLogistic}}(s) = (1+\exp(-c \cdot s))^{-1}$	$\mathcal{T}_{\text{Logistic}}(s)^{\#} = \frac{\tau}{2}\cosh^{-2}(\tau s),\ \tau = \frac{\pi}{2\sqrt{3}\sigma}$	$\mathcal{T}_{\text{Student}}(s)^{\S} = \frac{\Gamma\left(\frac{\nu+1}{2}\right)}{\Gamma\left(\frac{\nu}{2}\right)\sqrt{\pi\nu\sigma^2}}\left(1+\frac{s^2}{\nu\sigma^2}\right)^{-\frac{\nu+1}{2}}$

$^{\#}$ variance is σ^2, $\quad ^{\S}$ variance is $\frac{\nu}{\nu-2}\sigma^2$ for $\nu > 2$



(a) Bernoulli for classification (b) Gaussian and logistic (c) Laplacian and Student's t

(d) log of Bernoulli for classification (e) log of Gaussian and logistic (f) log of Laplacian and Student's t

Figure 2.2: Super-Gaussian potentials
Non-exhaustive list of usable potentials $\mathcal{T}(s)$ and their analytical expressions. We have the Gaussian potential for regression, the logistic, Laplace and Student's t potentials for robust regression and the cumulative Gaussian and logistic for classification. We also show the potentials in the log domain to make their asymptotic tail behaviour apparent: Laplace, logistic and cumulative logistic are asymptotically linear, Gaussian and cumulative Gaussian are asymptotically quadratic and Student's t has logarithmic asymptotics in the log domain.

Nowadays, inference engines such as Infer.NET [Minka et al., 2009] offer convenient access to general purpose inference code. However, the fully connected structure of our GLMs makes efficient inference in large models difficult for such a general solver because explicit awareness of the specific model structure leads to substantial computational benefits.

2.5.3 Approximations to achieve tractability

Bayesian inference is appealing from a conceptual point of view; however, there are many algorithmical challenges when attempting a tractable implementation on a computer. Therefore, approximations have to be made at various stages to achieve tractability.

Formal tractability means that beliefs about the model can be cast into a probability distribution at all. Very often the choice of model is guided by the available distributions; the often used term *convenience prior* criticises specific prior choices because they are often selected due to their usability.

In the following, the motivation for approximations is twofold: on the one hand there is the problem of *analytical intractability* of posterior distributions leading to approximations based on tractable distributions. On the other hand there will be the problem of *computational intractability* if the size of the inference problem is too big to enable efficient computations.

Analytical tractability can be achieved by representing the posterior by a member $\mathcal{Q}_{\varsigma}(\mathbf{u})$ of a tractable parametric family of distributions parametrised by ς. The most important families include delta distributions $\mathbb{Q}_\varsigma(\mathbf{u}) = \delta(\mathbf{u} - \hat{\mathbf{u}}) = \lim_{\epsilon \to 0} \mathcal{N}(\mathbf{u}|\hat{\mathbf{u}}, \epsilon\mathbf{I})$ with $\boldsymbol{\varsigma} = \hat{\mathbf{u}}$, factorial distributions $\mathbb{Q}_\varsigma(\mathbf{u}) = \prod_i \mathbb{Q}_{\varsigma_i}(u_i)$ and Gaussian distributions $\mathbb{Q}_\varsigma(\mathbf{u}) = \mathcal{N}(\mathbf{u}|\mathbf{m}, \mathbf{V})$ with $\boldsymbol{\varsigma} = (\mathbf{m}, \mathbf{V})$. They facilitate the calculation of expectations, and thus enable analytical decision making.

Computational tractability is a problem if large amounts of data have to be processed, even with an analytically tractable model. Two solutions are possible: approximate computations or dataset subsampling. We will examine approximate inference in linear Gaussian models in detail in the next section, where standard methods from linear algebra are used to approximate the posterior.

All algorithms discussed in the following are – of course – formally tractable, however only the Gaussian case is analytically tractable. Even though, all methods are computationally tractable, they differ in how the computational effort scales with growing number of potentials $p = n + q$. The critical point in achieving scalability with p is whether one can formulate the algorithm based on few evaluations of efficient primitives such as fast matrix vector multiplications (MVM) with the matrices $\mathbf{X}$ and $\mathbf{B}$. This is possible in the Gaussian model (section 2.5.4), for conjugate gradient (CG) approaches to MAP inference (section 2.5.6), using a fixed-point for factorial approximations (section 2.5.7 and Miskin [2000]). Proper variational approaches are harder to handle using few MVMs only. Expectation propagation (section 2.5.10) and KL-divergence minimisation (section 2.5.8) require many of them. Only the variational relaxation (section 2.5.9 and chapter 3) allows a decomposition of the objective so that approximate inference becomes scalable.

2.5.4 The Gaussian linear model

Bayesian inference in the Gaussian linear model is analytically feasible, however for large numbers of variables $m + q$, the computations become computationally challenging due to the sheer size of the matrices $\mathbf{X}$ and $\mathbf{B}$. We use the setting of section 2.2.2 and wish to compute the posterior mean $\mathbf{m} = \mathbb{E}_{\mathbb{P}(\mathbf{u}|\mathcal{D})}[\mathbf{u}]$ and its covariance $\mathbf{V} = \mathbb{V}_{\mathbb{P}(\mathbf{u}|\mathcal{D})}[\mathbf{u}]$, where

$$\mathbf{m} = \mathbf{A}^{-1}\mathbf{X}^\top\mathbf{y},\ \mathbf{V} = \sigma^2\mathbf{A}^{-1},\ \mathbf{A} = \mathbf{X}^\top\mathbf{X} + \mathbf{B}\boldsymbol{\Gamma}^{-1}\mathbf{B}.$$

The mean $\mathbf{m}$ is given by the solution of the linear system $\mathbf{Am} = \mathbf{b} := \mathbf{X}^\top\mathbf{y}$. Linear systems can be solved exactly by decomposing the matrix $\mathbf{A}$, e.g. by the Cholesky decomposition, which costs $\mathcal{O}(n^3)$. However, if n, the size of $\mathbf{A}$, becomes overly large, the computation becomes prohibitive; even the explicit storage of $\mathbf{A}$ is impossible if $n > 10^5$.

If $\mathbf{A}$ does not have further *exploitable structure*, we simply cannot compute the mean $\mathbf{m}$. We use "having exploitable structure" interchangeably with "enabling fast MVMs" – faster than $\mathcal{O}(n^2)$. Fast MVMs can be the consequence of $\mathbf{A}$ being sparse, a property that can be inherited from the system matrices $\mathbf{X}$ and $\mathbf{B}$ leading to a complexity of $\mathcal{O}(\#nz)$, i.e. linear in the number of nonzero elements in the matrix. Other exploitable structure exists if $\mathbf{X}$, $\mathbf{B}$ are members of special families of matrices such as Fourier matrices, finite derivative matrices or wavelet transform matrices having complexities $\mathcal{O}(n \cdot \ln n)$, $\mathcal{O}(n)$ and $\mathcal{O}(n)$, respectively.

Approximate mean computation with conjugate gradients

Computation of the mean $\mathbf{m}$ can alternatively be accomplished by the linear conjugate gradient algorithm (LCG) [Hestenes and Stiefel, 1952, Golub and van Loan, 1996, § 10.2]. Gaussian belief propagation (GBP) has been recognised as an instance of LCG. Derived as a sequential minimisation scheme of $f(\mathbf{m}) = \|\mathbf{Am} - \mathbf{b}\|_2^2$, where in each iteration, one MVM with $\mathbf{A}$ is needed to compute the next descent direction. Often, LCG needs far less than n iterations to converge, making it the method of choice for large matrices $\mathbf{A}$ with exploitable structure. The final computational cost is $\mathcal{O}(k \cdot v)$, where k is the number of MVMs needed and v is the cost of a single MVM with $\mathcal{O}(v) \geq \mathcal{O}(n)$.

Approximate variance computation with Lanczos

A much more difficult endeavour is the computation of the posterior covariance matrix $\mathbf{V}$, where sometimes only the diagonal $\mathrm{dg}(\mathbf{V})$ is of interest. In principle, the computation of $\mathbf{V}$ requires a matrix inversion, which is an $\mathcal{O}(n^3)$ process in general. We can compute rows $\mathbf{v}_i$ (or equivalently columns) of $\mathbf{V}$ by solving a linear system

$$\mathbf{v}_i = \mathbf{V}\mathbf{e}_i = \sigma^2\mathbf{A}^{-1}\mathbf{e}_i \Leftrightarrow \mathbf{A}\mathbf{v}_i = \sigma^2\mathbf{e}_i$$

leading to a prohibitive computational cost of $\mathcal{O}(k \cdot v \cdot n) \geq \mathcal{O}(k \cdot n^2)$ to compute all of $\mathbf{V}$. An approximate method [Schneider and Willsky, 2001] is based on the Lanczos algorithm [Lanczos, 1950, Golub and van Loan, 1996, § 9]. Used to compute eigenvector/eigenvalue pairs of large matrices, the Lanczos algorithm is a sequential procedure, requiring one MVM per iteration. The result of the Lanczos algorithm (after k iterations) is an orthogonal matrix $\mathbf{Q}_k \in \mathbb{R}^{n\times k}$ (i.e. $\mathbf{Q}_k^\top\mathbf{Q}_k = \mathbf{I} \in \mathbb{R}^{k\times k}$) tridiagonalising $\mathbf{A}$ so that $\mathbf{Q}_k^\top\mathbf{A}\mathbf{Q}_k = \mathbf{T}_k$ with tridiagonal $\mathbf{T}_k \in \mathbb{R}^{k\times k}$ whose eigenvector/eigenvalue pairs approach eigenvector/eigenvalue pairs of $\mathbf{A}$. From the Lanczos algorithm, we finally get an increasingly accurate low-rank approximation to $\mathbf{A}$ and $\mathbf{V}$ by

$$\mathbf{A} \approx \mathbf{Q}_k\mathbf{T}_k\mathbf{Q}_k^\top, \text{ and hence } \mathbf{V} \approx \sigma^2\mathbf{Q}_k\mathbf{T}_k^{-1}\mathbf{Q}_k^\top \text{ where } \mathbf{A} \succeq \mathbf{Q}_k\mathbf{T}_k\mathbf{Q}_k^\top \succeq \mathbf{Q}_{k-1}\mathbf{T}_{k-1}\mathbf{Q}_{k-1}^\top \succeq \mathbf{0}.$$

An undesirable feature of the Lanczos algorithm is the large storage requirements for the matrix $\mathbf{Q}_k$; we have to keep it in memory since every converged eigenvector induces a loss of orthogonality in $\mathbf{Q}_k$, which can be corrected by a Gram-Schmidt reorthogonalisation step necessitating access to the entire matrix $\mathbf{Q}_k$. The overall computational complexity is $\mathcal{O}(k \cdot v + k^3 \cdot n)$ and the required storage amounts to $\mathcal{O}(k \cdot n)$.

Even though we have discussed approximate computations for the linear Gaussian model, we will use them as building blocks inside approximate inference computations in non-Gaussian models (chapter 3.5.4).

2.5.5 Variational framework for non-Gaussian models

At the very core of Bayesian inference is the problem of computing high-dimensional integrals. Most often, these computations are only feasible for special distributions such as factorial or Gaussian distributions. Therefore, the most successful approach to approximate Bayesian inference in large continuous models is variational calculus [e.g. Jordan et al., 1999]. Optimisation problems (especially convex ones) are routinely solved at very large scales in numerical mathematics and machine learning, which lead to a variety of efficient algorithms. Exploiting that experience, a *variational algorithm* solves an (approximately) equivalent optimisation problem instead of the original problem.

How can we phrase the computation of posterior moments as an optimisation problem? Starting from a parametric family of distributions $\mathbb{Q}_\varsigma(\mathbf{u})$, where the moment computations are simple, we can pick the parameter

$$\varsigma^\star = \arg\min_\varsigma D(\mathbb{P}||\mathbb{Q}_\varsigma)$$

so that $\mathbb{Q}_{\varsigma^\star}(\mathbf{u})$ captures the most relevant properties of the posterior $\mathbb{P}(\mathbf{u}|\mathcal{D})$ via an optimisation w.r.t. ς. All algorithms discussed in the sequel (and many more) are instances of the divergence measure and message passing framework by Minka [2005], where global similarity or closeness between $\mathbb{P}(\mathbf{u}|\mathcal{D})$ and its approximation $\mathbb{Q}_\varsigma(\mathbf{u})$ is measured by the α-divergence

$$D_\alpha(\mathbb{P}||\mathbb{Q}_\varsigma) := \frac{1}{\alpha - \alpha^2}\left(1 - \int \left[\frac{\mathbb{P}(\mathbf{u}|\mathcal{D})}{\mathbb{Q}_\varsigma(\mathbf{u})}\right]^\alpha \mathbb{Q}_\varsigma(\mathbf{u})\mathrm{d}\mathbf{u}\right).$$

The α-divergence is non-negative, definite and convex in its arguments $\mathbb{P}$ and $\mathbb{Q}_\varsigma$. Two limiting cases for $\alpha = 0, 1$

$$\lim_{\alpha\to 1} D_\alpha(\mathbb{P}||\mathbb{Q}_\varsigma) = \mathrm{KL}(\mathbb{P}||\mathbb{Q}_\varsigma), \lim_{\alpha\to 0} D_\alpha(\mathbb{P}||\mathbb{Q}_\varsigma) = KL(\mathbb{Q}_\varsigma||\mathbb{P})$$

are especially important since they correspond to the Kullback-Leibler (KL) divergence

$$\mathrm{KL}(\mathbb{Q}_\varsigma||\mathbb{P}) := \int \mathbb{Q}_\varsigma(\mathbf{u}) \ln \frac{\mathbb{Q}_\varsigma(\mathbf{u})}{\mathbb{P}(\mathbf{u}|\mathcal{D})} \mathrm{d}\mathbf{u} = -\mathcal{H}\left[\mathbb{Q}_\varsigma\right] - \int \mathbb{Q}_\varsigma(\mathbf{u}) \ln \mathbb{P}(\mathbf{u}|\mathcal{D}) \mathrm{d}\mathbf{u}. \tag{2.9}$$

The KL-divergence is not symmetric; therefore swapping the arguments changes the objective. Whereas, $\mathrm{KL}(\mathbb{P}||\mathbb{Q}_\varsigma)$ is minimised by the Gaussian approximation $\mathbb{Q}_\varsigma$ having the same moments as $\mathbb{P}$, the minimisation of $\mathrm{KL}(\mathbb{Q}_\varsigma||\mathbb{P})$ is qualitatively different. Since the average is w.r.t. $\mathbb{Q}_\varsigma$ instead of $\mathbb{P}$, (see equation 2.9), the approximation can "choose" where it "wants to be" most accurate. However, if $\mathbb{P}(\mathbf{u}) = 0$, the KL-divergence is infinite unless $\mathbb{Q}_\varsigma(\mathbf{u}) = 0$ as well. This, so called *zero forcin*g property, enforces that $\mathbb{Q}_\varsigma$ and $\mathbb{P}$ agree in their respective support. One consequence of zero forcing is *mode seeking* meaning that a unimodal approximation to $\mathbb{P}$ has the tendency to approximate the mass around the mode.

In the next sections, we will discuss several approximate inference algorithms applied to the generalised linear model, each corresponding to a particular choice of α and $\mathbb{Q}_{\boldsymbol{\theta}}$ as summarised in table 2.3. All considered algorithms are deterministic approximations in contrast to Markov Chain Monte Carlo (MCMC) and other sampling approaches.

Name	Short	α	$\mathbb{Q}_\varsigma$	Criterion	Other name or equivalent algorithm
Laplace's method	LA	0	$\delta(\mathbf{u}-\hat{\mathbf{u}})$	local	Taylor expansion around the mode
Factorial variational approximation	FV	0	$\prod_i \mathbb{Q}_{\varsigma_i}(u_i)$	global	Mean field approximation
Gaussian KL minimisation	KL	0	$\mathcal{N}(\mathbf{u}\vert\mathbf{m},\mathbf{V})$	global	Structured mean field, Jensen bounding
Individual variational potential bounding	VB	0	$\mathcal{N}(\mathbf{u}\vert\mathbf{m},\mathbf{V})$	global	Constrained KL or integrand bounding
Expectation propagation	EP	1	$\mathcal{N}(\mathbf{u}\vert\mathbf{m},\mathbf{V})$	both	ADATAP or EC

Table 2.3: Properties of approximate inference algorithms

All methods have their respective way of computing an approximation or bound to the marginal likelihood $\mathbb{P}(\mathcal{D})$ with the following relations

$$\ln Z_{EP} \approx \ln Z \approx \ln Z_{LA}, \; \ln Z \geq \ln Z_{FV}, \; \ln Z \geq \ln Z_{KL} \geq \ln Z_{VB}$$

between them. In addition to that, for models agreeing in their marginals, the EC marginal likelihood dominates the variational bound [Opper and Winther, 2005, 3.1]

$$\ln Z_{EP} \geq \max(\ln Z_{FV}, \ln Z_{KL}).$$

We will see that VB is a special case of KL with lots of desirable properties. Except for the FV method, all approaches yield a Gaussian approximation to the posterior $\mathbb{P}(\mathbf{u}|\mathcal{D})$. All algorithms except for LA are focusing on *global* properties of the posterior; LA looks at the *local* height and curvature of $\mathbb{P}(\mathbf{u}|\mathcal{D})$ only. EP is doing both. Furthermore, KL can be understood as an *average* version of LA.

In chapter 3, we analyse the VB objective in detail and derive a scalable algorithm for its minimisation. In chapter 4, we empirically reformulate all approximation schemes for the case of Gaussian process classification as outlined in section 2.4. Later, in chapter 5, we use EP to drive experimental design to optimise image measurement architectures for small images. Finally, in chapter 6, we use the scalable VB algorithm of chapter 3 to optimise the measurement architecture for magnetic resonance imaging for medical images of realistic sizes.

Properties of the posterior

Depending on the potentials used, the posterior

$$\mathbb{P}(\mathbf{u}|\mathcal{D}) = \frac{C_{\mathcal{N}} C_{\mathcal{T}}}{Z} \mathcal{N}(\mathbf{y}|\mathbf{X}\mathbf{u}, \sigma^2\mathbf{I}) \prod_{j=1}^{q} \mathcal{T}_j(\mathbf{b}_j^\top \mathbf{u}) \tag{2.10}$$

will have different qualitative properties. The two most interesting properties for our investigations are *log-concavity* and *super-Gaussianity*.

A potential is log-concave if $g(s) = \ln \mathcal{T}(s)$ is a concave function or equivalently $-g(s)$ is a convex function.

$$f(s) \text{ is convex if } f(\lambda s + (1-\lambda)t) \leq \lambda f(s) + (1-\lambda) f(t) \; \forall \lambda \in [0,1] \forall s,t \in \mathbb{R}, s \neq t.$$

In other words, there is a slope α and an offset β so that

$$g(s) \leq \alpha s + \beta, \; \forall s$$

meaning we can find a linear upper bound on the log potential. A direct consequence of log-concave potentials is a unimodal posterior $\mathbb{P}(\mathbf{u}|\mathcal{D})$ rendering MAP estimation a convex minimisation problem. All potentials in figure 2.2 except for Student's t are log-concave.

A potential $\mathcal{T}(s)$ is strongly super-Gaussian if $g(x) = \ln \mathcal{T}(s)$, $x = s^2$ is strictly convex and non-increasing for $x > 0$ [Palmer et al., 2006]. As a consequence, $\mathcal{T}(s)$ can be lower-bounded by a centred Gaussian for any given variance γ up to a log-linear term e^{bs}

$$\exists b \in \mathbb{R} \forall \gamma \in \mathbb{R}_+ \forall s \in \mathbb{R} : \; \ln \mathcal{T}(s) + bs \overset{c}{\geq} \ln \mathcal{N}(s|0,\gamma).$$

Intuitively, the logarithm of strongly super-Gaussian functions can be lower bounded by a quadratic function. All potentials in figure 2.2 except for the Gaussian and the cumulative Gaussian are strongly super-Gaussian. The two exceptions have quadratic asymptotics causing the lower bounds to exist only up to a certain variance given by the asymptotics. However all potentials of figure 2.2 are super-Gaussian meaning that their tails are at least as heavy as a Gaussian tail. There are also non-super-Gaussian, i.e. *sub-Gaussian* potentials, e.g. potentials with bounded support are sub-Gaussian. In statistics, super-Gaussian is equivalent to *leptokurtic,* i.e. having a positive *kurtosis*.

If a potential $\mathcal{T}(s)$ is super-Gaussian and log-concave (all except Student's t), we can informally say that the logarithm of the potential $\mathcal{T}(s)$ is somewhere in between a linear and a quadratic function and equivalently that the potential $\mathcal{T}(s)$ is between the Gaussian and the Laplace distribution.

Marginal likelihood bound and KL-divergence

The marginal likelihood Z can be lower bounded using Jensen's inequality

$$\begin{aligned} \ln Z &= \ln C_{\mathcal{N}} C_{\mathcal{T}} + \ln \int \mathbb{Q}(\mathbf{u}) \frac{\mathcal{N}(\mathbf{y}|\mathbf{X}\mathbf{u}, \sigma^2\mathbf{I}) \prod_{j=1}^q \mathcal{T}_j(s_j)}{\mathbb{Q}(\mathbf{u})} \mathrm{d}\mathbf{u} \qquad (2.11) \\ &= \ln C_{\mathcal{N}} C_{\mathcal{T}} + \max_{\mathbb{Q}(\mathbf{u})} \int \mathbb{Q}(\mathbf{u}) \ln \frac{\mathcal{N}(\mathbf{y}|\mathbf{X}\mathbf{u}, \sigma^2\mathbf{I}) \prod_{j=1}^q \mathcal{T}_j(s_j)}{\mathbb{Q}(\mathbf{u})} \mathrm{d}\mathbf{u} \\ &\overset{\text{Jensen}}{\geq} \ln C_{\mathcal{N}} C_{\mathcal{T}} + \int \mathbb{Q}(\mathbf{u}) \ln \frac{\mathcal{N}(\mathbf{y}|\mathbf{X}\mathbf{u}, \sigma^2\mathbf{I}) \prod_{j=1}^q \mathcal{T}_j(s_j)}{\mathbb{Q}(\mathbf{u})} \mathrm{d}\mathbf{u} := \ln Z_B. \end{aligned}$$

We can recognise the term $\ln Z_B$ also in the KL-divergence

$$\begin{aligned} \mathrm{KL}(\mathbb{Q}||\mathbb{P}) &= \ln C_{\mathcal{N}} C_{\mathcal{T}} + \int \mathbb{Q}(\mathbf{u}) \ln \frac{\mathcal{N}(\mathbf{y}|\mathbf{X}\mathbf{u}, \sigma^2\mathbf{I}) \prod_{j=1}^q \mathcal{T}_j(s_j)}{\mathbb{Q}(\mathbf{u})} \mathrm{d}\mathbf{u} - \ln Z \\ &= \ln Z_B - \ln Z \qquad (2.12) \end{aligned}$$

and conclude that they are intimately connected. In variational approximations, one can equivalently minimise $\mathrm{KL}(\mathbb{Q}_\varsigma||\mathbb{P})$ with respect to variational parameters or maximise a corresponding lower bound on the marginal likelihood $\ln Z_B(\varsigma)$.

2.5.6 Laplace's method

The computationally simplest approach to approximate inference consists of a second order Taylor expansion of $\ln \mathbb{P}(\mathbf{u}|\mathcal{D})$ at its maximum $\hat{\mathbf{u}}$, which corresponds to a Gaussian approximation at the mode and where $\frac{\partial \ln \mathbb{P}(\hat{\mathbf{u}}|\mathcal{D})}{\partial \mathbf{u}} = \mathbf{0}$. Formally, we have

$$\begin{aligned}\hat{\mathbf{u}} &= \arg\max_{\mathbf{u}} \mathbb{P}(\mathbf{u}|\mathcal{D}) = \arg\min_{\mathbf{u}} \frac{1}{2\sigma^2} \|\mathbf{X}\mathbf{u} - \mathbf{y}\|^2 - \sum_{j=1}^{q} \ln \mathcal{T}_j(s_j), \ \mathbf{s} = \mathbf{B}\mathbf{u} \\ \ln \mathbb{P}(\mathbf{u}|\mathcal{D}) &\overset{c}{\approx} -\frac{1}{2}(\mathbf{u} - \hat{\mathbf{u}})^\top \mathbf{V}^{-1}(\mathbf{u} - \hat{\mathbf{u}}), \ \mathbf{V}^{-1} = -\frac{\partial^2 \ln \mathbb{P}(\hat{\mathbf{u}}|\mathcal{D})}{\partial \mathbf{u} \partial \mathbf{u}^\top} = \sigma^{-2}\left(\mathbf{X}^\top\mathbf{X} + \mathbf{B}^\top \mathbf{\Gamma}^{-1}\mathbf{B}\right),\end{aligned}$$

where $\gamma_j^{-1}\sigma^2 = \frac{\mathrm{d}}{\mathrm{d}s_j} \ln \mathcal{T}_j(s_j)$. This immediately suggests the IRLS algorithm of section 2.3.1 to solve the MAP problem. Of course, the method is most sensible for unimodal posteriors. Also, the covariance only depends on the curvature of the log posterior at the mode making it an approximation that is only locally justified. The optimisation of $\ln \mathbb{P}(\mathbf{u}|\mathcal{D})$ is a convex program if all potentials are log-concave.

The algorithm can alternatively be interpreted from a variational perspective using the KL-divergence and the set of delta distributions centred at $\hat{\mathbf{u}}$ as approximating family e.g. $\mathbb{Q}_{\hat{\mathbf{u}}}(\mathbf{u}) = \delta(\mathbf{u} - \hat{\mathbf{u}}) = \lim_{\epsilon \to 0} \mathcal{N}(\mathbf{u}|\hat{\mathbf{u}}, \epsilon\mathbf{I})$. Minimisation of the KL-divergence

$$\begin{aligned}\mathrm{KL}(\mathbb{Q}_{\hat{\mathbf{u}}}||\mathbb{P}) &= -\mathcal{H}\left[\lim_{\epsilon\to 0} \mathcal{N}(\mathbf{u}|\hat{\mathbf{u}}, \epsilon\mathbf{I})\right] - \int \delta(\mathbf{u} - \hat{\mathbf{u}}) \ln \mathbb{P}(\mathbf{u}|\mathcal{D}) \mathrm{d}\mathbf{u} \\ &= -\mathcal{H}\left[\delta(\mathbf{u})\right] - \ln \mathbb{P}(\hat{\mathbf{u}}|\mathcal{D})\end{aligned}$$

can be understood as the maximisation of the posterior because the differential entropy of the delta distribution $\delta(\mathbf{u})$ does not depend on the variational parameter $\hat{\mathbf{u}}$. However, the differential entropy $\mathcal{H}\left[\delta(\mathbf{u})\right]$ approaches $-\infty$ as ε goes to zero which renders the KL-divergence a rather useless measure.

Marginal likelihood

An approximation to the marginal likelihood can be obtained by also considering the posterior value at the mode $\mathbb{P}(\hat{\mathbf{u}}|\mathcal{D})$

$$\begin{aligned}\ln Z &= \ln C_{\mathcal{N}} C_{\mathcal{T}} + \ln \int \mathcal{N}(\mathbf{y}|\mathbf{X}\mathbf{u}, \sigma^2\mathbf{I}) \prod_{j=1}^{q} \mathcal{T}_j(s_j) \mathrm{d}\mathbf{u} \\ &\approx \ln C_{\mathcal{N}} C_{\mathcal{T}} + \frac{1}{2} \ln |\mathbf{V}| + \ln \mathcal{N}(\mathbf{y}|\mathbf{X}\hat{\mathbf{u}}, \sigma^2\mathbf{I}) + \sum_{j=1}^{q} \ln \mathcal{T}_j(\mathbf{b}_j^\top \hat{\mathbf{u}}) := \ln Z_{LA} \\ &= \ln C_{\mathcal{N}} C_{\mathcal{T}} + \frac{n-m}{2} \ln \sigma^2 - \frac{1}{2} \ln |\mathbf{A}| - \frac{1}{2\sigma^2} \|\mathbf{X}\hat{\mathbf{u}} - \mathbf{y}\|^2 + \sum_{j=1}^{q} \ln \mathcal{T}_j(\mathbf{b}_j^\top \hat{\mathbf{u}}),\end{aligned}$$

where $\mathbf{A} = \mathbf{X}^\top\mathbf{X} + \mathbf{B}^\top\mathbf{\Gamma}^{-1}\mathbf{B}$.

Computational complexity

The minimisation using IRLS or CG is efficient and scales well with the number of potentials p. Marginal likelihood computations are intrinsically harder since the exact evaluation of the $\ln |\mathbf{A}|$ term is cubic in p. However, the Lanczos approach of section 2.5.4 allows computing bounds.

2.5.7 Factorial variational approximation

A variational approach very commonly used in physics [Chandler, 1987, Parisi, 1988], is the mean field approximation, where the posterior $\mathbb{P}(\mathbf{u}|\mathcal{D})$ is approximated by the closest factorial distribution $\prod_{i=1}^{n} \mathbb{Q}_i(u_i)$ as measured by the KL-divergence. We derive the functional form of that distribution using variational calculus to find the optimal lower bound on the marginal likelihood, which is equivalent to minimising the KL-divergence (equation 2.12)

$$\begin{aligned}
\ln Z &\geq \ln C_{\mathcal{N}} C_{\mathcal{T}} + \int \prod_{i=1}^{n} \mathbb{Q}(u_i) \ln \frac{\mathcal{N}(\mathbf{y}|\mathbf{X}\mathbf{u}, \sigma^2 \mathbf{I}) \prod_{j=1}^{q} \mathcal{T}_j(s_j)}{\prod_{i=1}^{n} \mathbb{Q}(u_i)} \mathrm{d}\mathbf{u}, \; s_j = \mathbf{b}_j^\top \mathbf{u} \qquad (2.13) \\
&\overset{c}{=} \sum_{i=1}^{n} \mathcal{H}\left[\mathbb{Q}_i\right] - \frac{1}{2\sigma^2} \int \prod_{i=1}^{n} \mathbb{Q}_i(u_i) \left(\mathbf{u} \underbrace{\mathbf{X}^\top \mathbf{X}}_{\mathbf{D}} \mathbf{u} \underbrace{-2\mathbf{y}^\top \mathbf{X}}_{\mathbf{c}} \mathbf{u} \right) \mathrm{d}\mathbf{u} + \sum_{j=1}^{q} \int \prod_{i=1}^{n} \mathbb{Q}_i(u_i) \ln \mathcal{T}_j(s_j) \mathrm{d}\mathbf{u} \\
\ln Z_{FV} &:= \ln C_{\mathcal{N}} C_{\mathcal{T}} + \sum_{i=1}^{n} \mathcal{H}\left[\mathbb{Q}_i\right] - \frac{\mathbf{m}^\top \mathbf{D}\mathbf{m} + \mathbf{v}^\top \mathrm{dg}(\mathbf{D}) + \mathbf{c}^\top \mathbf{m}}{2\sigma^2} + \sum_{j=1}^{q} \int \prod_{i=1}^{n} \mathbb{Q}_i(u_i) \ln \mathcal{T}_j(s_j) \mathrm{d}\mathbf{u} \\
\text{where} \quad & m_i := \int \mathbb{Q}_i(u_i) u_i \mathrm{d}u_i, \quad v_i := \int \mathbb{Q}_i(u_i)(u_i - m_i)^2 \mathrm{d}u_i = \int \mathbb{Q}_i(u_i) u_i^2 \mathrm{d}u_i - m_i^2.
\end{aligned}$$

Then using

$$\frac{\delta \mathcal{H}\left[\mathbb{Q}_i\right]}{\delta \mathbb{Q}_i}(u_i) = -\ln \mathbb{Q}_i(u_i) - 1, \quad \tfrac{\delta m_i}{\delta \mathbb{Q}_i}(u_i) = u_i, \text{ and } \quad \frac{\delta v_i}{\delta \mathbb{Q}_i}(u_i) = (u_i - m_i)^2,$$

we can compute the functional derivative and set it to 0 to be able to read off the optimal form

$$\begin{aligned}
\frac{\delta \ln Z_{FV}}{\delta \mathbb{Q}_i}(u_i) &= -\ln \mathbb{Q}_i(u_i) - 1 - \frac{d_{ii}(u_i - m_i)^2 + \left(2\mathbf{m}^\top \mathrm{dg}(\mathbf{D}) + c_i\right) u_i}{2\sigma^2} + \ln \tilde{\mathcal{T}}_i(u_i) \\
& \frac{\delta \ln Z_{FV}}{\delta \mathbb{Q}_i}(u_i) \equiv 0 \Rightarrow \mathbb{Q}_i(u_i) = \tilde{Z}_i^{-1} \mathcal{N}(u_i|\tilde{\mu}_i, \tilde{\sigma}_i^2) \tilde{\mathcal{T}}_i(u_i), \qquad (2.14)
\end{aligned}$$

where the substitute potential $\tilde{\mathcal{T}}_i(u_i) = \exp\left[\sum_{j=1}^{q} \int \ln \mathcal{T}(\mathbf{b}_j^\top \mathbf{u}) \prod_{k \neq i} (\mathbb{Q}_k(u_k) \mathrm{d}u_k)\right]$ is a complicated function of u_i. Only for $\mathbf{B} = \mathbf{I}$, the expression simplifies considerably into $\tilde{\mathcal{T}}_i(u_i) = \mathcal{T}_i(u_i)$.

Knowing the functional form of the posterior approximation, we can optimise $\ln Z_{FV}$ with respect to the variational parameters $(\tilde{\boldsymbol{\mu}}, \tilde{\boldsymbol{\Sigma}})$.

Marginal likelihood

We start from the general expression for $\ln Z_{FV}$ of equation 2.13 and plug in the functional form of $\mathbb{Q}(u_i)$. It turns out that the formula simplifies a lot for $\mathbf{B} = \mathbf{I}$ to yield

$$\begin{aligned}
\ln Z_{FV} &= \ln C_{\mathcal{N}} C_{\mathcal{T}} + \ln \int \prod_i \mathcal{T}(u_i) \mathcal{N}(\mathbf{u}|\tilde{\boldsymbol{\mu}}, \tilde{\boldsymbol{\Sigma}}) \ln \frac{\mathcal{N}(\mathbf{y}|\mathbf{X}\mathbf{u}, \sigma^2 \mathbf{I})}{\mathcal{N}(\mathbf{u}|\tilde{\boldsymbol{\mu}}, \tilde{\boldsymbol{\Sigma}})} \mathrm{d}\mathbf{u} \\
&= \ln C_{\mathcal{N}} C_{\mathcal{T}} - \sum_{i=1}^{n} \left(\ln \tilde{Z}_i^{-1} - \ln \tilde{\sigma}_i \right) - m \ln \sigma + \frac{n-m}{2} \ln 2\pi - \frac{\mathbf{m}^\top \mathbf{D}\mathbf{m} + \mathbf{v}^\top \mathrm{dg}(\mathbf{D}) + \mathbf{c}^\top \mathbf{m} + e}{2} \\
\text{with} \quad & m_i := \int \mathbb{Q}_i(u_i) u_i \mathrm{d}u_i, \quad v_i := \int \mathbb{Q}_i(u_i)(u_i - m_i)^2 \mathrm{d}u_i, \quad \mathbf{D} := \sigma^{-2} \mathbf{X}^\top \mathbf{X} + \tilde{\boldsymbol{\Sigma}}^{-1} \\
\text{and} \quad & \mathbf{c} = -2(\sigma^{-2} \mathbf{X}^\top \mathbf{y} + \tilde{\boldsymbol{\Sigma}}^{-1} \boldsymbol{\mu}), \; e = \sigma^{-2} \mathbf{y}^\top \mathbf{y} + \boldsymbol{\mu}^\top \tilde{\boldsymbol{\Sigma}}^{-1} \boldsymbol{\mu}.
\end{aligned}$$

Computational complexity

The maximisation of $\ln Z_{FV}$ using a fixed-point algorithm [Miskin, 2000] is efficient and scales very well with the number of potentials p. Marginal likelihood computations are cheap, due to the factorial approximation. The major drawback of the method is the fact that it cannot properly capture correlations between pairs of variables.

2.5.8 Gaussian KL minimisation

Here, we simply fit the closest Gaussian distribution $\mathbb{Q}_\varsigma(\mathbf{u}) = \mathcal{N}(\mathbf{u}|\mathbf{m},\mathbf{V})$ in terms of the KL-divergence $\mathrm{KL}(\mathbb{Q}_\varsigma||\mathbb{P})$ to the posterior $\mathbb{P}(\mathbf{u}|\mathcal{D})$ [Opper and Archambeau, 2009] – a model with $\frac{n}{2}(n+3)$ parameters. Again, we start from Jensen's lower bound on the marginal likelihood

$$\begin{aligned}
\ln Z_{KL} &= \ln C_\mathcal{N} C_\mathcal{T} + \int \mathcal{N}(\mathbf{u}|\mathbf{m},\mathbf{V}) \ln \frac{\mathcal{N}(\mathbf{y}|\mathbf{X}\mathbf{u},\sigma^2\mathbf{I}) \prod_{j=1}^q \mathcal{T}_j(s_j)}{\mathcal{N}(\mathbf{u}|\mathbf{m},\mathbf{V})} \mathrm{d}\mathbf{u} \qquad (2.15)\\
&= C + \frac{1}{2}\ln|\mathbf{V}| - \frac{1}{2\sigma^2}\mathrm{tr}\left(\mathbf{X}\mathbf{V}\mathbf{X}^\top\right) - \frac{1}{2\sigma^2}\|\mathbf{X}\mathbf{m}-\mathbf{y}\|^2 + \sum_{j=1}^q \int \mathcal{N}(s)\ln \mathcal{T}_j(\sigma_j s + \mu_j)\mathrm{d}s\\
C &:= \ln C_\mathcal{N} C_\mathcal{T} + \frac{n}{2}(1+\ln 2\pi) - \frac{m}{2}\ln(2\pi\sigma^2),\ \mu_j = \mathbf{b}_j^\top \mathbf{m},\text{ and } \sigma_j = \sqrt{\mathbf{b}_j^\top \mathbf{V}\mathbf{b}_j}.
\end{aligned}$$

By equating the derivative

$$\begin{aligned}
\frac{\partial \ln Z_{KL}}{\partial \mathbf{V}} &= \frac{1}{2}\mathbf{V}^{-1} - \frac{1}{2\sigma^2}\mathbf{X}^\top\mathbf{X} + \mathbf{B}^\top \left[\int \frac{\partial \mathcal{N}(s|\mu_j,\sigma_j^2)}{\partial \sigma_j^2} \ln \mathcal{T}_j(s)\mathrm{d}s\right]_{jj} \mathbf{B} \overset{!}{=} \mathbf{0}\\
\Leftrightarrow\ \mathbf{V}^{-1} &= \frac{1}{\sigma^2}(\mathbf{X}^\top\mathbf{X} + \mathbf{B}^\top\mathbf{\Gamma}^{-1}\mathbf{B}),\ \gamma_j = -\frac{1}{\sigma^2}\int \frac{\partial \mathcal{N}(s|\mu_j,\sigma_j^2)}{\partial \sigma_j^2}\ln \mathcal{T}_j(s)\mathrm{d}s
\end{aligned}$$

with zero, we find that the covariance at the optimum is of the form

$$\mathbf{V}^\star = \sigma^2\mathbf{A}^{-1},\ \mathbf{A} = \mathbf{X}^\top\mathbf{X} + \mathbf{B}^\top\mathbf{\Gamma}^{-1}\mathbf{B} \qquad (2.16)$$

implying that $\mathbf{V}$ has only q parameters γ_j [Seeger, 2003] instead of $\frac{n}{2}(n+1)$ parameters V_{ij}. As a consequence, we can – without loss of generality – use Gaussian potential approximations $\tilde{\mathcal{T}}_j(s_j) = \exp\left(\frac{\beta_j}{\sigma^2}s_j - \frac{1}{2\sigma^2\gamma_j}s_j^2\right) \propto \mathcal{N}\left(s_j|\beta_j\gamma_j,\sigma^2\gamma_j\right)$ with $\varsigma = (\boldsymbol{\beta},\boldsymbol{\gamma})$; the optimum is the same as if using a full Gaussian with $\varsigma = (\mathbf{m},\mathbf{V})$. In fact, we compute the equivalent or effective Gaussian potential for every non-Gaussian potential. That means, we will use the approximate posterior

$$\mathbb{Q}_\varsigma(\mathbf{u}) = \frac{1}{\tilde{Z}}\mathcal{N}(\mathbf{y}|\mathbf{X}\mathbf{u},\sigma^2\mathbf{I})\prod_{j=1}^q \tilde{\mathcal{T}}_j(s_j),\ \tilde{Z} = \int \mathcal{N}(\mathbf{y}|\mathbf{X}\mathbf{u},\sigma^2\mathbf{I})\prod_{j=1}^q \tilde{\mathcal{T}}_j(s_j)\mathrm{d}\mathbf{u} \qquad (2.17)$$

instead of $\mathbb{Q}_\varsigma(\mathbf{u}) = \mathcal{N}(\mathbf{u}|\mathbf{m},\mathbf{V})$. Opper and Archambeau [2009] show that the fixed point conditions at the mode $\hat{\mathbf{u}}$ for Laplace's method

$$\frac{\partial \ln \mathbb{P}(\hat{\mathbf{u}}|\mathcal{D})}{\partial \mathbf{u}} = \mathbf{0} \quad\text{and}\quad \mathbf{V}^{-1} = -\frac{\partial^2 \ln \mathbb{P}(\hat{\mathbf{u}}|\mathcal{D})}{\partial \mathbf{u}\partial \mathbf{u}^\top}$$

hold on average for the KL methods since

$$\mathbb{E}_{\mathcal{N}(\mathbf{u}|\mathbf{m},\mathbf{V})}\left[\frac{\partial \ln \mathbb{P}(\hat{\mathbf{u}}|\mathcal{D})}{\partial \mathbf{u}}\right] = \mathbf{0} \quad\text{and}\quad \mathbf{V}^{-1} = \mathbb{E}_{\mathcal{N}(\mathbf{u}|\mathbf{m},\mathbf{V})}\left[-\frac{\partial^2 \ln \mathbb{P}(\hat{\mathbf{u}}|\mathcal{D})}{\partial \mathbf{u}\partial \mathbf{u}^\top}\right].$$

Intuitively, the marginal likelihood bound of equation 2.15 seen as a function of the mean $\mathbf{m}$ at the optimum $\mathbf{V}^\star$ is a smoothed version (with smoothing width $\sigma_j^\star$) of the MAP objective of section 2.5.6.

$$\begin{aligned}
\ln Z_{KL}(\mathbf{m}) &\overset{c}{=} -\frac{1}{2\sigma^2}\|\mathbf{X}\mathbf{m}-\mathbf{y}\|^2 + \sum_{j=1}^q \int \mathcal{N}\left(s|\mathbf{b}_j^\top\mathbf{m},(\sigma_j^\star)^2\right)\ln \mathcal{T}_j(s)\mathrm{d}s\\
&= -\frac{1}{2\sigma^2}\|\mathbf{X}\mathbf{m}-\mathbf{y}\|^2 + \sum_{j=1}^q \int \mathcal{N}(s|0,1)\ln \mathcal{T}_j\left(\sigma_j^\star s + \mathbf{b}_j^\top\mathbf{m}\right)\mathrm{d}s\\
\ln \mathbb{P}(\mathbf{u}|\mathcal{D}) &\overset{c}{=} -\frac{1}{2\sigma^2}\|\mathbf{X}\mathbf{u}-\mathbf{y}\|^2 + \sum_{j=1}^q \ln \mathcal{T}_j(\mathbf{b}_j^\top\mathbf{u})
\end{aligned}$$

Most notably, if $-\ln \mathbb{P}(\mathbf{u}|\mathcal{D})$ is convex (e.g. all potentials are log-concave) then $-\ln Z_{KL}(\mathbf{m})$ is also convex since weighted sums preserve convexity [Boyd and Vandenberghe, 2004, §3.2.1]. However, $-\ln Z_{KL}(\boldsymbol{\gamma})$ is not convex in general since

$$-\ln Z_{KL}(\mathbf{V}) \overset{c}{=} -\frac{1}{2}\ln|\mathbf{V}| + \frac{1}{2\sigma^2}\mathrm{tr}\left(\mathbf{X}\mathbf{V}\mathbf{X}^\top\right) + \sum_{j=1}^{q}\omega_j(\mu_j,\sigma_j^2), \quad \sigma_j^2 = \mathbf{b}_j^\top\mathbf{V}\mathbf{b}_j,\ \mu_j = \mathbf{m}^\top\mathbf{b}_j$$

is in general not convex in $\mathbf{V} = \sigma^2(\mathbf{X}^\top\mathbf{X} + \mathbf{B}^\top\boldsymbol{\Gamma}^{-1}\mathbf{B})^{-1}$ let alone in $\boldsymbol{\Gamma}$. In appendix C.8, we show that – even though $\omega_j(\mu_j,\sigma_j^2) = -\int \mathcal{N}(s)\ln\mathcal{T}_j(\sigma_j s + \mu_j)\,\mathrm{d}s$ is in general jointly convex in (μ_j,σ_j) – it is at least not convex in σ_j^2 for Laplace potentials.

Marginal likelihood

Plugging the $(\boldsymbol{\beta},\boldsymbol{\gamma})$ parametrisation into the lower bound, we obtain the alternative expression

$$\begin{aligned}
\ln Z_{KL} &= \ln C_{\mathcal{N}}C_{\mathcal{T}} + \ln\tilde{Z} + \sum_{j=1}^{q}\int \mathrm{Q}_{\boldsymbol{\varsigma}}(\mathbf{u})\ln\frac{\mathcal{T}_j(s_j)}{\tilde{\mathcal{T}}_j(s_j)}\mathrm{d}\mathbf{u} \qquad (2.18)\\
\ln\tilde{Z} &= \int \mathcal{N}(\mathbf{y}|\mathbf{X}\mathbf{u},\sigma^2\mathbf{I})\prod_{j=1}^{q}\tilde{\mathcal{T}}_j(s_j)\mathrm{d}\mathbf{u} = \frac{n-m}{2}\ln(2\pi\sigma^2) + \frac{\mathbf{m}^\top\mathbf{A}\mathbf{m} - \mathbf{y}^\top\mathbf{y}}{2\sigma^2} - \frac{1}{2}\ln|\mathbf{A}|\\
&= \frac{n-m}{2}\ln(2\pi\sigma^2) - \frac{1}{2\sigma^2}\min_{\mathbf{u}}\left(\|\mathbf{X}\mathbf{u}-\mathbf{y}\|^2 + \mathbf{s}^\top\boldsymbol{\Gamma}^{-1}\mathbf{s} - 2\boldsymbol{\beta}^\top\mathbf{s}\right) - \frac{1}{2}\ln|\mathbf{A}|,
\end{aligned}$$

where we used the shorthands $\mathbf{A} = \mathbf{X}^\top\mathbf{X} + \mathbf{B}^\top\boldsymbol{\Gamma}^{-1}\mathbf{B}$, $\mathbf{d} = \mathbf{X}^\top\mathbf{y} + \mathbf{B}^\top\boldsymbol{\beta}$ and $\mathbf{m} = \mathbf{A}^{-1}\mathbf{d}$. The step in the last line is based on the relationship $-\mathbf{c}^\top\mathbf{A}^{-1}\mathbf{c} = \min_{\mathbf{x}}\mathbf{x}^\top\mathbf{A}\mathbf{x} - 2\mathbf{c}^\top\mathbf{x}$ (see appendix B); we use it only because it will appear in chapter 3. Approximate inference is done by maximising $\ln Z_{KL}$ with respect to the variational parameters $\boldsymbol{\varsigma} = (\boldsymbol{\beta},\boldsymbol{\gamma})$.

Computational complexity

The maximisation of $\ln Z_{KL}$ using a quasi Newton [Opper and Archambeau, 2009] or Newton algorithm [Nickisch and Rasmussen, 2008] and the marginal likelihood evaluation do not scale well with the number of potentials p due to cubic matrix operations. Using the Lanczos procedure from section 2.5.4, we can indeed approximately compute gradients of $\ln Z_{KL}$. However, there is no cheap way to select the step size (since $\ln Z_{KL}$ is hard to evaluate), which is crucial for gradient-based methods to properly converge.

2.5.9 Individual variational potential bounding

A closely related algorithm is based on the idea to individually lower bound every non-Gaussian potential $\mathcal{T}_j(s_j)$ by a scaled parametrised Gaussian and to maximise the marginal likelihood with respect to these parameters [Jaakkola and Jordan, 1996, Gibbs and MacKay, 2000, Palmer et al., 2006]. We call the method VB to emphasise the variational lower bounds. Formally, one uses a lower bound

$$\mathcal{T}_j(s_j) \geq \exp\left(\frac{\beta_j(\gamma_j)}{\sigma^2}s_j - \frac{1}{2\sigma^2\gamma_j}s_j^2 - \frac{h_j(\gamma_j)}{2}\right) = \hat{\mathcal{T}}_j(s_j;\gamma_j) \propto \mathcal{N}\left(s_j|\beta_j\gamma_j,\sigma^2\gamma_j\right) \qquad (2.19)$$

with width parameter γ_j to derive a lower evidence bound

$$\begin{aligned}
\ln Z &= \ln C_{\mathcal{N}}C_{\mathcal{T}} + \ln\int \mathcal{N}(\mathbf{y}|\mathbf{X}\mathbf{u},\sigma^2\mathbf{I})\prod_{j=1}^{q}\mathcal{T}_j(s_j)\mathrm{d}\mathbf{u}\\
&= \ln C_{\mathcal{N}}C_{\mathcal{T}} + \ln\int \mathcal{N}(\mathbf{y}|\mathbf{X}\mathbf{u},\sigma^2\mathbf{I})\max_{\boldsymbol{\gamma}}\prod_{j=1}^{q}\hat{\mathcal{T}}_j(s_j;\gamma_j)\mathrm{d}\mathbf{u}\\
&\geq \ln C_{\mathcal{N}}C_{\mathcal{T}} + \ln\int \mathcal{N}(\mathbf{y}|\mathbf{X}\mathbf{u},\sigma^2\mathbf{I})\prod_{j=1}^{q}\hat{\mathcal{T}}_j(s_j;\gamma_j)\mathrm{d}\mathbf{u} =: \ln Z_{VB}(\boldsymbol{\gamma}).
\end{aligned}$$

The relation to the KL method of section 2.5.8 is interesting: the individual potential bounding approach is a special case of the KL algorithm, where the parameter $\boldsymbol{\beta}$ becomes a function of γ and the lower bound $\ln Z_{VB}(\gamma)$ is a relaxation of $\ln Z_{KL}$. Comparing the Gaussian potential approximation of the KL method $\tilde{\mathcal{T}}_j(s_j)$, $\tilde{\mathcal{T}}(\mathbf{s}) = \prod_{j=1}^{q} \tilde{\mathcal{T}}_j(s_j)$ and the lower bound used in the VB method $\hat{\mathcal{T}}_j(s_j;\gamma_j)$, we find

$$\tilde{Z} = \int \mathcal{N}(\mathbf{y}|\mathbf{Xu},\sigma^2\mathbf{I})\tilde{\mathcal{T}}(\mathbf{s})\mathrm{d}\mathbf{u},\; \hat{\mathcal{T}}_j(s_j;\gamma_j) = \tilde{\mathcal{T}}_j(s_j;\gamma_j)\exp\left(-\frac{h_j(\gamma_j)}{2}\right) \Leftrightarrow \ln\frac{\hat{\mathcal{T}}_j(s_j;\gamma_j)}{\tilde{\mathcal{T}}_j(s_j;\gamma_j)} = -\frac{h_j(\gamma_j)}{2}$$

a relation that can be used to further lower bound $\ln Z_{KL}$ by

$$\begin{aligned}
\ln Z \geq \max_{\boldsymbol{\beta},\boldsymbol{\gamma}} \ln Z_{KL}(\boldsymbol{\beta},\boldsymbol{\gamma}) &= \max_{\boldsymbol{\beta},\boldsymbol{\gamma}} \left(\ln \tilde{Z} C_{\mathcal{N}} C_{\mathcal{T}} + \sum_{j=1}^{q} \int \mathbb{Q}_{\varsigma}(\mathbf{u}) \ln \frac{\mathcal{T}_j(s_j)}{\tilde{\mathcal{T}}_j(s_j)} \mathrm{d}\mathbf{u}\right) \\
&\overset{\boldsymbol{\beta}=\boldsymbol{\beta}(\boldsymbol{\gamma})}{\geq} \max_{\boldsymbol{\gamma}} \left(\ln \tilde{Z} C_{\mathcal{N}} C_{\mathcal{T}} + \sum_{j=1}^{q} \int \mathbb{Q}_{\varsigma}(\mathbf{u}) \ln \frac{\hat{\mathcal{T}}_j(s_j;\gamma_j)}{\tilde{\mathcal{T}}_j(s_j)} \mathrm{d}\mathbf{u}\right) \\
&= \max_{\boldsymbol{\gamma}} \left(\ln \tilde{Z} C_{\mathcal{N}} C_{\mathcal{T}} - \frac{1}{2}\sum_{j=1}^{q} h_j(\gamma_j)\right) = \max_{\boldsymbol{\gamma}} \ln Z_{VB}(\boldsymbol{\gamma}). \quad (2.20)
\end{aligned}$$

From the definition of $\ln \tilde{Z}$ in equation 2.17 and the bound of equation 2.19, we can see that the last line, indeed equals $\ln Z_{VB}$.

We will see in chapter 3, that $\ln Z_{VB}(\gamma)$ has very advantageous analytical and algorithmic properties leading to scalable and efficient algorithms. We will show that $-\ln Z_{VB}(\gamma)$ is a convex function if all potentials of the model are log-concave and super-Gaussian – a property that will theoretically corroborate the algorithm and practically simplify the variational optimisation.

Computational complexity

As we will discuss in chapter 3, $\ln Z_{VB}$ can be decoupled so that an efficient optimisation becomes possible. Furthermore, for super-Gaussian and log-concave potentials $\mathcal{T}(s_j)$, this leads to a convex minimisation problem.

2.5.10 Expectation propagation

The expectation propagation algorithm [Minka, 2001a] generalises loopy belief propagation (LBP)[Frey and MacKay, 1998, Murphy et al., 1999] from the machine learning literature and assumed density filtering (ADF) [Maybeck, 1982] from the control literature and is equivalent to approaches from statistical physics such as adaptive TAP (ADATAP) [Opper and Winther, 2000] and the expectation consistency (EC) framework by Opper and Winther [2005].

EP attempts to globally minimise the KL-divergence (with average computed w.r.t. $\mathbb{P}(\mathbf{u}|\mathcal{D})$)

$$\mathrm{KL}\left(\mathbb{P}\|\,\mathbb{Q}\right) = \int \mathbb{P}(\mathbf{u}|\mathcal{D}) \ln \frac{\mathbb{P}(\mathbf{u}|\mathcal{D})}{\mathbb{Q}(\mathbf{u})} \mathrm{d}\mathbf{u}$$

between the exact posterior $\mathbb{P}(\mathbf{u}|\mathcal{D}) = C_{\mathcal{N}} C_{\mathcal{T}} \mathcal{N}(\mathbf{y}|\mathbf{Xu},\sigma^2\mathbf{I}) \prod_{j=1}^{q} \mathcal{T}_j(s_j)$ and the Gaussian

$$\mathbb{Q}(\mathbf{u}) = C_{\mathcal{N}} C_{\mathcal{T}} \mathcal{N}(\mathbf{y}|\mathbf{Xu},\sigma^2\mathbf{I}) \prod_{j=1}^{q} \tilde{\mathcal{T}}_j(s_j),\; \tilde{\mathcal{T}}_j(s_j) = Z_j^{-1} \exp\left(\frac{\beta_j}{\sigma^2} s_j - \frac{1}{2\sigma^2\gamma_j} s_j^2\right) \propto \mathcal{N}\left(s_j|\beta_j\gamma_j,\sigma^2\gamma_j\right),$$

where all potentials $\mathcal{T}_j(s_j)$ have been replaced by scaled Gaussians $\tilde{\mathcal{T}}_j(s_j)$ acting as the Gaussian equivalents of $\mathcal{T}_j(s_j)$. Since the global integration in $\mathrm{KL}\left(\mathbb{P}||\mathbb{Q}\right)$ over all q non-Gaussian potentials jointly is analytically intractable, the minimisation is relaxed to considering one non-Gaussian potential at a time and hence to 1d integrations over s_j

$$\mathrm{KL}\left(\mathbb{Q}(\mathbf{u})\frac{\mathcal{T}_j(s_j)}{\tilde{\mathcal{T}}_j(s_j)}\middle\|\,\mathbb{Q}(\mathbf{u})\right)=\mathrm{KL}\left(\mathbb{Q}_j(s_j)\frac{\mathcal{T}_j(s_j)}{\tilde{\mathcal{T}}_j(s_j)}\middle\|\,\mathbb{Q}_j(s_j)\right)=\mathrm{KL}\left(\mathbb{Q}_{\neg j}\cdot\mathcal{T}_j\|\,\mathbb{Q}_{\neg j}\cdot\tilde{\mathcal{T}}_j\right) \quad (2.21)$$

using Gaussian marginals $\mathbb{Q}(s_j)=\mathcal{N}(s_j|\mathbf{b}_j^\top\mathbf{m},\mathbf{b}_j^\top\mathbf{V}\mathbf{b}_j)$ of the approximation $\mathbb{Q}(\mathbf{u})=\mathcal{N}(\mathbf{u}|\mathbf{m},\mathbf{V})$ to the exact posterior. Here, the (unnormalised) *cavity distributions* $\mathbb{Q}_{\neg j}(s_j):=\mathbb{Q}_j(s_j)/\tilde{\mathcal{T}}_j(s_j)$ summarise the contextual information about s_j contained in the approximate posterior if the approximate potential $\tilde{\mathcal{T}}_j(s_j)$ is removed. The local KL-divergence in equation 2.21 is minimised if the variational parameters (Z_j,β_j,γ_j) of $\tilde{\mathcal{T}}_j(s_j)$ are chosen so that $\mathbb{Q}_{\neg j}(s_j)\mathcal{T}_j(s_j)$ and $\mathbb{Q}_{\neg j}(s_j)\tilde{\mathcal{T}}_j(s_j)$ have the same moments

$$\int s_j^k\mathbb{Q}_{\neg j}(s_j)\tilde{\mathcal{T}}_j(s_j)\mathrm{d}s_j \;=\; \int s_j^k\mathbb{Q}_{\neg j}(s_j)\mathcal{T}_j(s_j)\mathrm{d}s_j,\; k=0,1,2. \quad (2.22)$$

Algorithmically, these local moment matching steps are iterated over $j=1,..,q$ until convergence. The fixed point of the EP algorithm is a saddle point of the EP marginal likelihood [Minka, 2005] of equation 2.23 or equivalently the *EC free energy* [Opper and Winther, 2005]. Note that by using the KL-divergence the "other way round" $\mathrm{KL}\left(\mathbb{Q}_{\neg j}\cdot\tilde{\mathcal{T}}_j\|\,\mathbb{Q}_{\neg j}\cdot\mathcal{T}_j\right)$, we obtain a local updating scheme (variational message passing [Winn and Bishop, 2005]) minimising the global objective [Minka, 2005] of the KL method of section 2.5.8.

To summarise, EP can be understood in three ways: first, EP is an algorithm iterating local updates based on moment matching. Second, it is a fixed point of a free energy function and third, it is a system of nonlinear equations (equation 2.22).

Marginal likelihood

Replacing the potentials $\mathcal{T}_j(s_j)$ by their equivalent Gaussians $\tilde{\mathcal{T}}_j(s_j)$ in equation 2.11 and using the definition for $\tilde{Z}$ in equation 2.17, we obtain the EP marginal likelihood

$$\ln Z_{Ep} \;=\; \ln C_{\mathcal{N}}C_{\mathcal{T}}+\ln\int\mathcal{N}(\mathbf{y}|\mathbf{X}\mathbf{u},\sigma^2\mathbf{I})\prod_{j=1}^q\tilde{\mathcal{T}}_j(s_j)\mathrm{d}\mathbf{u}=\ln C_{\mathcal{N}}C_{\mathcal{T}}+\ln\tilde{Z}-\sum_{j=1}^q\ln Z_j (2.23)$$

that shares the term $\ln C_{\mathcal{N}}C_{\mathcal{T}}+\ln\tilde{Z}$ with $\ln Z_{KL}$ (equation 2.18) and $\ln Z_{VB}$ (equation 2.20). Note that $\ln Z_{EP}\geq\ln Z_{KL}\geq\ln Z_{VB}$ and $\ln Z_{EP}\geq\ln Z_{FV}$ if the algorithms yield the same marginals [Opper and Winther, 2005, 3.1].

Computational complexity

Every EP update step requires access to the posterior marginals $\mathbb{Q}_j(s_j)$. To compute one marginal exactly, one has to solve a linear system of size n (section 2.5.4). So, every sweep through all q potentials is at least of quadratic complexity $\mathcal{O}(q\cdot n)$, which is prohibitive. The problem is the sequential updating one-by-one in contrast to a gradient step updating all potentials jointly. For Gaussian process models (section 4.4), linear systems require $\mathcal{O}(n^3)$ in general, therefore EP implementations keep a representation of the posterior covariance of size $\mathcal{O}(n^2)$ either by storing $\mathbf{V}$ or equivalently some Cholesky factor of the same size [Rasmussen and Williams, 2006, ch. 3.6.3] in order to guarantee $\mathcal{O}(1)$ access to the marginals. Furthermore, EP requires numerically exact calculations to properly converge [Seeger, 2008, p. 773] rendering approximations less attractive.

2.6 Experimental design

Experimental design allows to guide the measurement process itself in order to acquire only the most informative data points $(\mathbf{x}_i,y_i)$. Often, the data matrix $\mathbf{X}$ containing the covariates is simply called the *design matrix*.

The frequentist or classical experimental design methodology as introduced by Fisher [1935] tries to decrease the variance of the estimator $\hat{\mathbf{u}}$ for the unknown variables $\mathbf{u}$. As a result, the design criteria are based on the eigenvalues of the estimator's covariance matrix or lower bounds thereof. Modern books on the subject include Atkinson and Donev [2002], Pukelsheim [2006].

The Bayesian approach is different since the unknown $\mathbf{u}$ is treated as a random variable with a prior $\mathbb{P}(\mathbf{u})$. Here, the goal is to reduce the entropy in the posterior $\mathbb{P}(\mathbf{u}|\mathbf{y})$. For a seminal review of Bayesian experimental design, see Chaloner and Verdinelli [1995].

As we will see, for the Gaussian linear model, Bayesian experimental design is equivalent to D-optimal frequentist design. However, for more complex models, the two approaches are very different. One distinction is that the Bayesian design score depends on the measurements $\mathbf{y}$ made so far, whereas only expectations w.r.t. the likelihood $\mathbb{P}(\mathbf{y}|\mathbf{u})$ appear in the frequentist score.

2.6.1 Frequentist experimental design

The basic frequentist idea is to select new data $(\mathbf{x}_*, y_*)$ so that the variance $\mathbf{V} = \mathbb{V}[\hat{\mathbf{u}}]$ of the estimator $\hat{\mathbf{u}} = \hat{\mathbf{u}}(\mathbf{X}, \mathbf{y})$ for the unknown $\mathbf{u}$ decreases as much as possible, where the particular choice of estimator determines the compromise between bias and variance. Most of the classical design criteria are p-norms of the vector $\boldsymbol{\lambda}$

$$\phi(\hat{\mathbf{u}}) \quad = \quad \|\boldsymbol{\lambda}\|_p = \left(\sum_{i=1}^{n} \lambda_i^p\right)^{\frac{1}{p}}, \ \lambda_i = \lambda_i(\mathbf{V})$$

whose components are the eigenvalues of $\mathbf{V}$ – a way to express the "size" of the matrix $\mathbf{V}$ as a scalar. Table 2.4 summarises the most common cost functions used in experimental design.

name of the design criterion	p	cost function $\phi(\hat{\mathbf{u}})$	intuition
D-optimality	0	$\prod_{i=1}^{n} \lambda_i = \lvert\mathbf{V}\rvert$	generalised variance
A-optimality	1	$\sum_{i=1}^{n} \lambda_i = \text{tr}(\mathbf{V})$	average variance
E-optimality	∞	$\max_i \lambda_i = \|\mathbf{V}\|_\infty$	maximal variance

Table 2.4: Experimental design cost functions

For the simple OLS estimator, we can analytically compute the variance, but for non-Gaussian likelihoods or more complicated estimators, it can be impossible to explicitly derive the variance. Using the likelihood $\mathbb{P}(\mathbf{y}|\mathbf{u})$, a distribution over $\mathbf{y}$ for fixed $\mathbf{u}$, the Cramér-Rao lower bound (CRB) [Cramér, 1946, Rao, 1945] on the variance of any estimator $\hat{\mathbf{u}}$ has the form

$$\mathbf{V} = \mathbb{V}[\hat{\mathbf{u}}] \succcurlyeq \frac{\partial \boldsymbol{\psi}}{\partial \mathbf{u}^\top} \mathbf{F}^{-1} \frac{\partial \boldsymbol{\psi}^\top}{\partial \mathbf{u}}, \ \boldsymbol{\psi} = \int \hat{\mathbf{u}} \mathbb{P}(\mathbf{y}|\mathbf{u}) d\mathbf{y}, \ \mathbf{F} = \int \frac{\partial \ln \mathbb{P}(\mathbf{y}|\mathbf{u})}{\partial \mathbf{u}} \frac{\partial \ln \mathbb{P}(\mathbf{y}|\mathbf{u})}{\partial \mathbf{u}^\top} \mathbb{P}(\mathbf{y}|\mathbf{u}) d\mathbf{y}, \tag{2.24}$$

where $\mathbf{F}$ is the *Fisher information matrix* and $\boldsymbol{\psi} = \mathbb{E}[\hat{\mathbf{u}}]$ is the expected value of the estimator under the likelihood. The bound is asymptotically tight for the maximum likelihood estimator. Often, *unbiased* estimators are used, where $\mathbb{E}[\hat{\mathbf{u}}] = \boldsymbol{\psi} = \mathbf{u}$ and hence $\mathbb{V}[\hat{\mathbf{u}}] \succcurlyeq \mathbf{F}^{-1}$. Since $\mathbf{V}$ does not have a closed form for many interesting models, one replaces $\mathbf{V}$ by its lower bound according to equation 2.24. For general likelihoods $\mathbb{P}(\mathbf{y}|\mathbf{u})$, also the expectation in the Fisher matrix is likely to be analytically intractable. Besides the CRB, there exists a big variety of lower bounds on $\mathbb{V}[\hat{\mathbf{u}}]$ [Bhattacharyya, 1946, Barankin, 1949, Abel, 1993] being sometimes tighter but more tedious to compute. For non-linear Gaussian models, the estimator's expectation $\mathbb{E}[\hat{\mathbf{u}}]$ is hard to compute. Further, for Gaussian likelihood $\mathbb{P}(\mathbf{y}|\mathbf{u}) = \mathcal{N}(\mathbf{y}|\mathbf{X}\mathbf{u}, \sigma^2\mathbf{I})$, the Fisher information matrix is given by $\mathbf{F} = \frac{1}{\sigma^2}\mathbf{X}^\top\mathbf{X}$, which is rank deficient if $m < n$. This property renders the approach inapplicable in underdetermined settings. In PLS (section 2.2.1), for example, depending on γ^{-1}, $\boldsymbol{\psi}$ ranges between $\mathbb{E}_{\gamma=0}[\hat{\mathbf{u}}] = \mathbf{0} \preceq \mathbb{E}[\hat{\mathbf{u}}] \preceq \mathbf{u} = \mathbb{E}_{\gamma=\infty}[\hat{\mathbf{u}}]$ giving rise to different values of the bias $\mathbb{E}[\hat{\mathbf{u}}] - \mathbf{u}$. There is one critical issue concerning the design methodology: we minimise a lower bound on the variance, however theoretical guarantees for the validity of this procedure apply only to the asymptotic regime of many observations. The small sample regime is less well understood.

Note that the criteria $\phi_{D,A,E}(\hat{\mathbf{u}})$ do not depend on the actual measurements $\mathbf{y}$ made so far; they are expectations w.r.t. $\mathbf{y}$ under the likelihood.

2.6.2 Bayesian experimental design

In Bayesian design philosophy, the unknown $\mathbf{u}$ is considered a random variable. A natural measure of uncertainty contained in a random variable $\mathbf{z}$ is its *(differential) entropy* [Cover and Thomas, 2006]

$$\mathcal{H}\left[\mathbb{P}(\mathbf{z})\right] \quad = \quad -\int \mathbb{P}(\mathbf{z}) \ln \mathbb{P}(\mathbf{z}) \mathrm{d}\mathbf{z}.$$

For fixed mean and variance, a Gaussian has maximal entropy (appendix C) leading to the upper bound

$$\mathcal{H}\left[\mathbb{P}(\mathbf{z})\right] \leq \mathcal{H}\left[\mathcal{N}\left(\mathbf{z}|\mathbb{E}_{\mathbb{P}(\mathbf{z})}[\mathbf{z}], \mathbb{V}_{\mathbb{P}(\mathbf{z})}[\mathbf{z}]\right)\right] = \frac{1}{2}\ln\left|\mathbb{V}_{\mathbb{P}(\mathbf{z})}[\mathbf{z}]\right| + \frac{n}{2}\left(1+\ln 2\pi\right), \, \mathbf{z} \in \mathbb{R}^n. \quad (2.25)$$

More accurate statements about the tightness of the bound are based on series approximations of $\mathbb{P}(\mathbf{z})$ as given in appendix D.4. Therefore, large variances are equivalent to high entropy implying very little information about the location of $\mathbf{z}$. At the core of the Bayesian design strategy is the idea to localise the posterior as much as possible. This is equivalent to decreasing the expected entropy of the posterior including the new data $\mathbf{x}_*$ relative to the entropy of the previous posterior without $\mathbf{x}_*$. Formally, we use the *information gain*

$$IG(\mathbf{x}_*) = \mathcal{H}[\mathbb{P}(\mathbf{u}|\mathbf{y})] - \int \mathcal{H}[\mathbb{P}(\mathbf{u}|\mathbf{y}, y_*)]\mathbb{P}(y_*|\mathbf{y})\mathrm{d}y_*, \quad (2.26)$$

where we need to compute the expected entropy $\mathcal{H}[\mathbb{P}(\mathbf{u}|\mathbf{y}, y_*)]$ of the augmented posterior including the measurement y_* along $\mathbf{x}_*$. The expectation is done over $\mathbb{P}(y_*|\mathbf{y}) = \int \mathbb{P}(\mathbf{u}|\mathbf{y})\mathbb{P}(y_*|\mathbf{u})\mathrm{d}\mathbf{u}$. Note that the information gain explicitly depends on the observations $\mathbf{y}$. In the applications of this thesis (see chapters 5&6), the integrals in equation 2.26 cannot be done analytically. Therefore, we will use approximate inference to replace $\mathbb{P}(\mathbf{u}|\mathbf{y})$ by $\mathbb{Q}(\mathbf{u})$ first with an approximation allowing an analytic computation of the information gain score. However, it is necessary to keep in mind that we approximate at various stages to obtain the design score: first, variational methods (except for EP) typically underestimate the posterior covariance and second the Gaussian entropy is an upper bound on the actual posterior entropy. As in case of frequentist design (section 2.6.1), theoretical results on the approximation quality are rare.

2.6.3 Information gain scores and approximate posteriors

For general posteriors $\mathbb{P}(\mathbf{u}|\mathbf{y})$ the information gain score $IG(\mathbf{X}_*)$ is analytically intractable. However, for Gaussian likelihoods $\mathbb{P}(\mathbf{y}_*|\mathbf{u}) = \mathcal{N}(\mathbf{y}_*|\mathbf{X}_*\mathbf{u}, \sigma^2\mathbf{I})$, we can use a Gaussian $\mathbb{Q}(\mathbf{u})$ to compute the information gain score $IG(\mathbf{X}_*)$ approximately. For non-Gaussian likelihoods, further approximations are necessary. With $\mathbb{P}(\mathbf{u}|\mathbf{y}, \mathbf{y}_*)\mathbb{P}(\mathbf{y}_*|\mathbf{y}) = \mathbb{P}(\mathbf{y}_*|\mathbf{y}, \mathbf{u})\mathbb{P}(\mathbf{u}|\mathbf{y})$ and $\mathbf{X}_* \in \mathbb{R}^{d\times n}, \mathbf{y}_* \in \mathbb{R}^d$, the score $IG(\mathbf{X}_*)$ can be expressed as the entropy of the new observations $\mathbf{y}_*$ given the old observations $\mathbf{y}$:

$$\begin{aligned} IG(\mathbf{X}_*) &= \mathcal{H}\left[\mathbb{P}(\mathbf{u}|\mathbf{y})\right] - \int \mathcal{H}\left[\mathbb{P}(\mathbf{u}|\mathbf{y}, \mathbf{y}_*)\right]\mathbb{P}(\mathbf{y}_*|\mathbf{y})\mathrm{d}\mathbf{y}_* \\ &= \mathcal{H}\left[\mathbb{P}(\mathbf{u}|\mathbf{y})\right] + \int\int \ln\left(\frac{\mathbb{P}(\mathbf{y}_*|\mathbf{y}, \mathbf{u})\mathbb{P}(\mathbf{u}|\mathbf{y})}{\mathbb{P}(\mathbf{y}_*|\mathbf{y})}\right)\mathbb{P}(\mathbf{u}|\mathbf{y}, \mathbf{y}_*)\mathrm{d}\mathbf{u}\mathbb{P}(\mathbf{y}_*|\mathbf{y})\mathrm{d}\mathbf{y}_* \\ &= \mathcal{H}\left[\mathbb{P}(\mathbf{u}|\mathbf{y})\right] + \int\int \ln \mathbb{P}(\mathbf{y}_*|\mathbf{y}, \mathbf{u})\mathbb{P}(\mathbf{u}|\mathbf{y}, \mathbf{y}_*)\mathbb{P}(\mathbf{y}_*|\mathbf{y})\mathrm{d}\mathbf{y}_*\mathrm{d}\mathbf{u} - \mathcal{H}\left[\mathbb{P}(\mathbf{u}|\mathbf{y})\right] + \mathcal{H}\left[\mathbb{P}(\mathbf{y}_*|\mathbf{y})\right] \\ &= \mathcal{H}\left[\mathbb{P}(\mathbf{y}_*|\mathbf{y})\right] - \int \mathcal{H}\left[\mathbb{P}(\mathbf{y}_*|\mathbf{u})\right]\mathbb{P}(\mathbf{u}|\mathbf{y})\mathrm{d}\mathbf{u} = \mathcal{H}\left[\mathbb{P}(\mathbf{y}_*|\mathbf{y})\right] - d\left(\frac{1}{2}\ln 2\pi e + \ln\sigma\right). \end{aligned}$$

Even though, $\mathbb{P}(\mathbf{y}_*|\mathbf{y})$ is a non-Gaussian distribution, its variance can be obtained by the law of total variance from the variance of the posterior $\mathbb{P}(\mathbf{u}|\mathbf{y})$

$$\begin{aligned}\mathbb{V}_{\mathbb{P}(\mathbf{y}_*|\mathbf{y})}\left[\mathbf{y}_*|\mathbf{y}\right] &= \mathbb{E}_{\mathbb{P}(\mathbf{u}|\mathbf{y})}\left[\mathbb{V}_{\mathbb{P}(\mathbf{y}_*|\mathbf{y},\mathbf{u})}\left[\mathbf{y}_*|\mathbf{y},\mathbf{u}\right]\right] + \mathbb{V}_{\mathbb{P}(\mathbf{u}|\mathbf{y})}\left[\mathbb{E}_{\mathbb{P}(\mathbf{y}_*|\mathbf{y},\mathbf{u})}\left[\mathbf{y}_*|\mathbf{y},\mathbf{u}\right]\right] \\ &= \mathbb{E}_{\mathbb{P}(\mathbf{u}|\mathbf{y})}\left[\sigma^2\mathbf{I}\right] + \mathbb{V}_{\mathbb{P}(\mathbf{u}|\mathbf{y})}\left[\mathbf{X}_*\mathbf{u}\right] \\ &= \sigma^2\mathbf{I} + \mathbf{X}_*\mathbb{V}_{\mathbb{P}(\mathbf{u}|\mathbf{y})}\left[\mathbf{u}\right]\mathbf{X}_*^\top.\end{aligned}$$

Using the Gaussian upper bound on the entropy (equation 2.25), we get a formula generalising the linear Gaussian case (equations 2.27 and 2.28) to

$$\begin{aligned}IG(\mathbf{X}_*) &\leq \frac{1}{2}\ln\left|\mathbb{V}_{\mathbb{P}(\mathbf{y}_*|\mathbf{y})}\left[\mathbf{y}_*\right]\right| + \frac{d}{2}\left(\ln 2\pi e\right) - d\left(\frac{1}{2}\ln 2\pi e + \ln\sigma\right) \\ &= \frac{1}{2}\ln\left|\mathbf{I} + \sigma^{-2}\mathbf{X}_*\mathbb{V}_{\mathbb{P}(\mathbf{u}|\mathbf{y})}\left[\mathbf{u}\right]\mathbf{X}_*^\top\right|.\end{aligned}$$

Since we seek for $\mathbf{X}_*$ with maximal information gain $IG(\mathbf{X}_*)$, the bound depends on the dominating eigenmodes of the posterior covariance matrix $\mathbb{V}_{\mathbb{P}(\mathbf{u}|\mathbf{y})}[\mathbf{u}]$. In applications where n is large and the approximate posterior covariance $\mathbf{V} = \mathbb{V}_{\mathbb{Q}(\mathbf{u})}[\mathbf{u}] = \sigma^2\left(\mathbf{X}^\top\mathbf{X} + \mathbf{B}^\top\mathbf{\Gamma}^{-1}\mathbf{B}\right)^{-1}$ cannot be stored as a dense matrix but is implicitly represented using MVMs with $\mathbf{X}$, $\mathbf{B}$ and the vector γ, the evaluation of $\mathbf{X}_*\mathbf{V}\mathbf{X}_*^\top$ is computationally demanding. Every row of $\mathbf{X}_*$ requires the solution of a linear system with the $n \times n$ matrix $\mathbf{V}$, which can – of course – be done by conjugate gradients . To alleviate this computational burden, one can use the Lanczos method of section 2.5.4 computing a low-rank approximation $\mathbf{V} \approx \sigma^2\mathbf{Q}_k\mathbf{T}_k^{-1}\mathbf{Q}_k^\top$. If the eigenmodes of $\mathbf{V}$ are well captured by the Lanczos approximation, we can expect the large score values to be rather accurate.

2.6.4 Constrained designs

Up to now, we require new measurement directions to have unit length $\mathrm{dg}(\mathbf{X}_*\mathbf{X}_*^\top) = \mathbf{1}$ otherwise, rescaling would always lead to an increase in information gain or equivalently a decrease in the estimator's variance. Further constraints might be present in practise. Most commonly, the rows of $\mathbf{X}_*$ can originate from a discrete set of candidates $\mathcal{X}_c$. In the so-called *transductive* setting [Yu et al., 2006], one has to find a discrete subset of the possible candidates rather than a continuous matrix. In general, the selection problem is of combinatorial complexity, however, there exist convex reformulations for the linear Gaussian case [Yu et al., 2008]. Unfortunately, they are useless in the underdetermined regime where $m < n$.

2.6.5 Sequential and joint designs

In the applications of this thesis, experimental design is not only used once. For complex design decisions based on data $(\mathbf{y}, \mathbf{X})$, we alternate in a loop between the inference step and the design decision for the next *single* $(y_*, \mathbf{x}_*)$ or *joint measurement* $(\mathbf{y}_*, \mathbf{X}_*)$ to include. Clearly, optimising a set of candidates $\mathbf{X}_*$ jointly can lead to better designs but is also computationally more demanding. Often, a greedy strategy will act as the pragmatic choice with only a single candidate $\mathbf{x}_*$ being added each time. The individual candidate measurements $\mathbf{x}_*$ can come from a discrete candidate set $\mathbf{x}_*^i$, $i \in I$ or from a continuous candidate space $\mathbf{x}_* \in \mathcal{X}$. In the former case, we simply select the candidate with highest score, and in the latter case, we have to optimise the design score w.r.t. $\mathbf{x}_*$ with gradient based methods, for example.

It is the inference step, that marks the difference between the frequentist and the Bayesian approach. In Frequentist design, we need to compute the inverse Fisher information matrix $\mathbf{F}_{\mathbf{x}_*}^{-1}$ for every candidate $\mathbf{x}_*$ and select the candidate with smallest cost ϕ. In Bayesian design, we compute an approximate posterior (basically a Gaussian) $\mathbb{Q}(\mathbf{u}) \approx \mathbb{P}(\mathbf{u}|\mathbf{y}, \mathbf{X})$ specifically tailored to facilitate the evaluation of the information gain score $IG(\mathbf{x}_*)$ and pick the candidate $\mathbf{x}_*$ yielding the biggest score.

On a higher level, the actual observations $\mathbf{y}$ and y_* do not enter the frequentist design loop as particular values; they are present through expectations only. In Bayesian methodology however, precisely these numbers form the basis for a proper assessment of the uncertainty left in the current state of knowledge about $\mathbf{u}$. In the regime of abundant data, $m \gg 0$, frequentist design is the method of choice since it implies a lot of asymptotic guarantees. However, in the underdetermined case $m < n$, the Bayesian approach is more appropriate as we will see in the following.

2.6.6 Bayesian versus frequentist design

D-optimal frequentist design and Bayesian experimental design based on a Gaussian approximation to the posterior distribution are similar in two ways: first, they both reduce uncertainty, i.e. either shrink the variance of the estimator or lower the posterior entropy, which is equivalent to decreasing the variance in a Gaussian approximation. Second, in the limit of many observations $m \to \infty$ and hence omission of the prior, they are the same. However, there are also severe differences: in the underdetermined case $m < n$, the frequentist approach is not applicable.

To make this more concrete, we have a look at the linear Gaussian case as detailed in section 2.2.1 and. For $p = 2$, the PLS estimator (equation 2.7) is given by $\hat{\mathbf{u}}_{\text{PLS}} = \mathbf{A}^{-1}\mathbf{X}^\top\mathbf{y}$ with $\mathbf{A} = \mathbf{X}^\top\mathbf{X} + \gamma^{-1}\mathbf{B}^\top\mathbf{B}$. Using the bilinearity of the covariance and $\mathbb{V}[\mathbf{y}] = \sigma^2\mathbf{I}$, we obtain the variance of the PLS estimator $\hat{\mathbf{u}}_{\text{PLS}}$

$$\hat{\mathbf{V}} := \mathbb{V}[\hat{\mathbf{u}}_{\text{PLS}}] \quad = \quad \mathbf{A}^{-1}\mathbf{X}^\top\mathbb{V}[\mathbf{y}]\mathbf{X}\mathbf{A}^{-1} = \sigma^2\mathbf{A}^{-1}\mathbf{X}^\top\mathbf{X}\mathbf{A}^{-1}.$$

Although, the PLS estimator coincides with the posterior mean, the posterior variance

$$\mathbf{V} := \mathbb{V}_{\mathbb{P}(\mathbf{u}|\mathcal{D})}[\mathbf{u}] \quad = \quad \sigma^2\mathbf{A}^{-1}$$

is distinctively different from $\hat{\mathbf{V}}$. As it will be shown in chapter 3, the diagonal $\boldsymbol{v} = \text{dg}(\mathbf{V})$ is bounded $\boldsymbol{v} \preceq \sigma^2\gamma\mathbf{1}$ from above by the prior variance, which does not hold for $\hat{\mathbf{V}}$. Also the rank of $\hat{\mathbf{V}}$ only depends on the rank of $\mathbf{X}^\top\mathbf{X}$. For underdetermined measurements $m < n$, $\hat{\mathbf{V}}$ inevitably becomes singular; it cannot be interpreted as the uncertainty of the current knowledge about $\mathbf{u}$ since it is impossible to achieve perfect certainty from a small number of noisy measurements.

Experimental design with D-optimality as criterion and invertible $\mathbf{X}^\top\mathbf{X}$, selects the next measurements $\mathbf{X}_* = [\mathbf{x}_{*,1}, .., \mathbf{x}_{*,d}]^\top$ to maximise the design score

$$\begin{aligned} -\ln\phi_D(\mathbf{X}_*, \hat{\mathbf{u}}_{\text{PLS}}) &= -\ln|\hat{\mathbf{V}}| = -\ln|\sigma^2(\mathbf{A}+\mathbf{X}_*\mathbf{X}_*^\top)^{-2}(\mathbf{X}^\top\mathbf{X}+\mathbf{X}_*\mathbf{X}_*^\top)|,\ \mathbf{A} = \mathbf{X}^\top\mathbf{X}+\boldsymbol{\Gamma}^{-1} \\ &\overset{c}{=} 2\ln|\mathbf{A}+\mathbf{X}_*\mathbf{X}_*^\top| - \ln|\mathbf{X}^\top\mathbf{X}+\mathbf{X}_*\mathbf{X}_*^\top| \\ &\overset{c}{=} 2\ln|\mathbf{I}+\mathbf{X}_*^\top\mathbf{A}^{-1}\mathbf{X}_*| - \ln|\mathbf{I}+\mathbf{X}_*^\top(\mathbf{X}^\top\mathbf{X})^{-1}\mathbf{X}_*|. \end{aligned} \tag{2.27}$$

The score compromises between choosing $\mathbf{X}_*$ along the biggest eigendirections of $\mathbf{A}^{-1}$ (Bayesian posterior variance) and along the smallest eigendirections of $(\mathbf{X}^\top\mathbf{X})^{-1}$ (OLS estimator variance).

The Bayesian information gain score

$$IG(\mathbf{x}_*) = -\frac{1}{2}\ln|\mathbf{A}| + \frac{1}{2}\ln\left|\mathbf{X}^\top\mathbf{X}+\mathbf{X}_*\mathbf{X}_*^\top+\boldsymbol{\Gamma}^{-1}\right| = \frac{1}{2}\ln\left|\mathbf{I}+\mathbf{X}_*^\top\mathbf{A}^{-1}\mathbf{X}_*\right|, \mathbf{A} = \mathbf{X}^\top\mathbf{X}+\boldsymbol{\Gamma}^{-1} \tag{2.28}$$

is equivalent to $-\ln\phi_D(\mathbf{X}_*, \hat{\mathbf{u}}_{\text{PLS}})$ in the flat prior limit $\boldsymbol{\Gamma} \to \infty \cdot \mathbf{I}$ only.

We use two toy examples with $n = 2$, $q = m = 1$ to illustrate the different behaviours: first let the measurement $\mathbf{X} = [0, 1]$ and the penalty domains $\mathbf{B} = [1, 0]$ be orthogonal $\mathbf{B}\mathbf{X}^\top = \mathbf{0} \in \mathbb{R}^{q \times m}$, hence

$$\mathbf{A} = \begin{pmatrix} \gamma^{-1} & 0 \\ 0 & 1 \end{pmatrix}, \hat{\mathbf{V}} = \sigma^2\begin{pmatrix} \gamma^2 & 0 \\ 0 & 0 \end{pmatrix} \Rightarrow \hat{\mathbf{x}}_* = \begin{pmatrix} 1 \\ 0 \end{pmatrix} \quad \text{and} \quad \mathbf{V} = \sigma^2\begin{pmatrix} \gamma & 0 \\ 0 & 1 \end{pmatrix}.$$

Thus, for $\gamma < 1$, the frequentist and Bayesian methodologies exactly suggest the opposite measurement; for larger prior variances, the Bayesian will measure u_1 as the frequentist. Note that in a sequential setting, the frequentist will always measure u_1 since he is absolutely certain about u_2.

Second, if $\mathbf{X} = [1,1]$, $\mathbf{B} = [1,0]$ we get

$$\begin{aligned}
\mathbf{A} &= \begin{pmatrix} 1+\gamma^{-1} & 1 \\ 1 & 1 \end{pmatrix}, \ \mathbf{A}^{-1} = \begin{pmatrix} \gamma & -\gamma \\ -\gamma & \gamma+1 \end{pmatrix} \\
\hat{\mathbf{V}} &= \sigma^2 \mathbf{A}^{-1}\mathbf{X}^\top \mathbf{X}\mathbf{A}^{-1} = \sigma^2 \begin{pmatrix} 0 & 0 \\ 0 & 1 \end{pmatrix} \Rightarrow \hat{\mathbf{x}}_* = \begin{pmatrix} 0 \\ 1 \end{pmatrix} \\
\mathbf{V} &= \sigma^2 \mathbf{A}^{-1} = \sigma^2 \begin{pmatrix} \gamma & -\gamma \\ -\gamma & \gamma+1 \end{pmatrix}.
\end{aligned}$$

Decomposing $\mathbf{A} = \mathbf{W}\mathbf{\Lambda}\mathbf{W}^\top$ with $\lambda = \frac{2+\gamma^{-1}\pm\sqrt{4+\gamma^{-2}}}{2}$ and $\mathbf{w} = \frac{1}{\sqrt{\lambda^2-2\lambda+2}} \begin{pmatrix} \lambda-1 \\ 1 \end{pmatrix}$ and using the smaller eigenvalue of $\mathbf{A}$, we can deduce $\mathbf{x}_* \in [-\xi; 1]$, where $\xi = \frac{1}{2}\sqrt{\gamma^{-2}+4} - \frac{1}{2}\gamma^{-1} \in [0,1]$.

2.7 Discussion and links to other chapters

Starting from a theoretical introduction into frequentist estimation and Bayesian inference in sections 2.1.1&2.1.2, we discussed the simplest possible application: the Gaussian linear model in section 2.2.

Two generalisations were addressed in terms of their respective estimation and inference procedures:

- Non-Gaussian noise leads to the generalised linear model (section 2.3). GLMs are used in the compressed image sensing application in chapter 5 and the magnetic resonance sequence design in chapter 6.
- Non-linear functional relationships can be achieved by Gaussian process models (section 2.4). Chapter 4 discusses applications of these non-linear model to classification.

In the technical section 2.5.4, we develop approaches to perform approximate Bayesian inference in a unified framework. After this broad overview, we explain, how the posterior approximation can be used to perform Bayesian experimental design in section 2.6.2. The frequentist design methodology is detailed in section 2.6.1.

Chapter 3 concentrates on one particular approximation method and sheds light on convexity properties and scalable optimisation algorithms. In chapter 4, we have a closer look at various aspects of all approximation methods in the context of Gaussian processes. Later, in chapter 5, we use expectation propagation to design the measurement architecture in an image acquisition task and finally, in chapter 6, we employ the algorithms of chapter 3 to optimise magnetic resonance trajectories.

Chapter 3

Convex Inference Relaxations and Algorithms

Point estimators are most often stated as the unique solution to an optimisation problem. Due to scalable optimisation algorithms, point estimators can be efficiently computed for models with very many variables. Approximate Bayesian inference is at its core a high-dimensional integration problem, which is computationally much harder to solve. Variational approaches, represent the integration as an optimisation problem to get access to the advanced algorithms making point estimation so efficient. However, typical variational problems are not only high-dimensional and strongly coupled; they enjoy much less analytically useful properties such as convexity.

In the following chapter, which is based on material contained in Nickisch and Seeger [2009] and Seeger and Nickisch [2008b], we discuss a particular variational inference method [Girolami, 2001, Palmer et al., 2006, Jaakkola, 1997] already mentioned in chapter 2.5.9. We provide convexity results, a scalable algorithm and experiments. The proposed inference algorithm is as scalable as the corresponding point estimation procedure that is contained as a special case.

In particular, we compare scale-mixture and variational bounding approaches to variational inference in sections 3.2 and 3.3, respectively to understand how non-Gaussian potentials can be represented by Gaussian ones. Then we derive convexity properties of the variational bounding technique in section 3.4 and provide an efficient optimisation algorithm in section 3.5 as well as a generic implementation in form of the `glm-ie` toolbox[1] (section 3.6). Finally, section 3.7 presents experimental results for an application to large scale binary classification active learning followed by a discussion in section 3.8.

3.1 Introduction

The class of models considered in the following comprises generalised linear models over continuous latent variables $\mathbf{u} \in \mathbb{R}^n$ with Gaussian $\mathcal{N}(r_i|y_i, \sigma^2)$ and non-Gaussian potentials $\mathcal{T}_j(s_j)$, where $\mathbf{r} = \mathbf{X}\mathbf{u}$ and $\mathbf{s} = \mathbf{B}\mathbf{u}$ (see figure 2.1).

For example, in the magnetic resonance imaging application of chapter 6, $\mathbf{u}$ denotes the unknown proton density image, $\mathbf{y} = \mathbf{X}\mathbf{u} + \boldsymbol{\varepsilon} \in \mathbb{C}^n$ are scanner measurements, where $\mathbf{X}$ is a Fourier sampling matrix, and the $\mathcal{T}_j(s_j)$ form a sparsity prior on multi scale image gradients s_j.

In binary classification (section 3.7), $\mathbf{u}$ correspond to classifier weights, $\mathbf{B}$ collects feature vectors $\mathbf{b}_j$ (or simply data points), and $\mathcal{T}_j(s_j)$ are cumulative logistic likelihoods. For a Gaussian prior on the weights $\mathbf{u}$, we have $\mathbf{X} = \mathbf{I}$ and $\mathbf{y} = \mathbf{0}$. However, a sparsity prior on the weights $\mathbf{u}$ leads to $\mathbf{X} = []$, $\mathbf{y} = []$, i.e. $m = 0$ Gaussian potentials; we have to append $\mathbf{I}$ to $\mathbf{B}$, and add sparsity potentials to the $\mathcal{T}_j(s_j)$.

The inference algorithm, we are discussing in this chapter provides a deterministic approximation to the posterior distribution of the model. Alternatively, sampling from high-

[1] `http://mloss.org/software/view/269/`

dimensional models is extremely challenging even though sophisticated samplers such as hybrid Monte Carlo techniques [Duane et al., 1987, Neal, 1993] are used. Proper estimation of posterior covariance modes, as needed for experimental design, is likely to require many samples from the posterior distribution. The Laplace approximation at the posterior mode (see chapter 2.5.6) is not applicable if non-differentiable potentials such as Laplace potentials are used because at the mode of such a model, the Hessian does not exist.

Our posterior approximation has a proper non-degenerate covariance enabling high-level tasks that rely on faithful approximations of uncertainty information (unrelated to the location of the posterior mode) such as experimental design, hyperparameter learning or feature relevance ranking. We show that our variational relaxation constitutes a convex optimisation problem, whenever the search for the posterior mode is convex. We propose an efficient double loop algorithm[2], reaching scalability by decoupling the criterion and reducing all efforts to standard techniques from numerical linear algebra. The algorithm is generically applicable to super Gaussian potentials and can be used in machine learning applications to infer good decisions from incomplete data, in settings with many unknown variables. Further, the algorithm allows to reliably operate Bayesian inference in large scale domains, where previously only convex point estimation techniques could be used. We show how our method applies to binary classification Bayesian active learning, with thousands of sequential inclusions.

Our algorithm is based on many convenient analytical properties of Gaussian models. Therefore, one way to attack inference in non-Gaussian models is to represent the non-Gaussian potentials $\mathcal{T}(s)$ by Gaussians $\mathcal{N}(s|0,\gamma)$ to exploit the simplicity of Gaussian computations. In the following, we will describe two prominent and related approaches: Gaussian scale mixtures (section 3.2) and variational bounds (section 3.3). They are applicable to a wide range of non-Gaussian potentials [Palmer et al., 2006] and naturally lead to a joint Gaussian approximation to the posterior distribution over the model. We will then concentrate on the variational bounding technique and its nice analytical properties leading to a scalable and efficient algorithm.

3.2 Gaussian scale mixtures and SBL

Gaussian *scale mixtures* allow to represent non-Gaussian potentials as a convex combination of Gaussians: consider a standard normal random variable $X \sim \mathcal{N}(0,1)$. The random variable $S := \theta + \sqrt{\gamma}X$, $\gamma > 0$ follows a Gaussian distribution $\mathcal{N}(\theta,\gamma)$ with *location parameter* θ and scale parameter $\sqrt{\gamma}$. If the parameters (θ,γ) have a joint density $\mathbb{P}(\theta,\gamma) = \mathbb{P}(\theta)\mathbb{P}(\gamma)$, independent of X, we can write

$$\mathbb{P}(s) = \int \int_0^\infty \mathcal{N}(s|\theta,\gamma)\mathbb{P}(\gamma)\mathbb{P}(\theta)\mathrm{d}\gamma\mathrm{d}\theta. \tag{3.1}$$

In general, the collection of all $S \sim \mathbb{P}(s)$ with a density of the form of equation 3.1 constitute the *location scale family* of the univariate random variable X, which covers a big class of univariate distributions. In table 3.1, a selection of prominent Gaussian scale mixtures are listed with their corresponding scale distribution. We look at zero-mean mixtures only, i.e. $\mathbb{P}(\theta) = \delta(\theta)$ allowing to represent non-Gaussian potentials by

$$\mathcal{T}(s) = \int_0^\infty \mathcal{N}(s|0,\gamma)\mathbb{P}(\gamma)d\gamma = \int_0^\infty \frac{\exp\left(-\frac{s^2}{2\gamma}\right)}{\sqrt{2\pi\gamma}}\mathbb{P}(\gamma)d\gamma = \int_0^\infty \tilde{\mathcal{T}}(s;\gamma)\frac{\mathbb{P}(\gamma)}{\sqrt{2\pi\gamma}}d\gamma,$$

where $\tilde{\mathcal{T}}(s;\gamma)$ denote the respective Gaussian potentials.

Sampling from $\mathbb{P}(s)$ is simple: first draw $\gamma \sim \mathbb{P}(\gamma)$, then draw $s \sim \mathcal{N}(0,\gamma)$.

Besides sampling, approximate inference can be done using the framework of sparse Bayesian learning (SBL) [Tipping, 2001]

[2] The MRI application from chapter 6 is contained as a special case.

#	Scale distribution $\mathbb{P}(\gamma) \propto$		Scale Mixture $\mathbb{P}(s) \propto$	
1)	Exponential, $\tau > 0$, $\mathcal{E}(\gamma\|\tau)$	$\frac{\tau^2}{2} \exp\left(-\frac{\tau^2}{2}\gamma\right)$	Laplace, $\mathcal{L}(s\|\tau)$	$\frac{\tau}{2} \exp\left(-\tau\|s\|\right)$
2)	Gamma on γ^{-1}, $\mathcal{G}(\gamma^{-1}\|\nu,\tau)$	$\gamma^{1-\nu/2} \exp\left(-\frac{\nu}{2\tau}\gamma^{-1}\right)$	$\nu = 2\alpha, \tau = \frac{\alpha}{\beta} > 0$, Student's t, $\mathcal{T}(s\|\nu)$	$\left(1+\frac{s^2}{\nu}\right)^{-\frac{\nu+1}{2}}$
3)	Infinite Gaussian mixture	$\sum_{n=1}^{\infty}(-1)^{n+1}n^2 \exp\left(-\frac{1}{2}n^2\gamma\right)$	Logistic $\mathcal{L}og(s)$	$\exp(-s) \cdot (\exp(-s)+1)^{-2}$
4)	Sym. stable, $\alpha \in (1,2)$, $\mathcal{SS}_\alpha(\gamma)$	$\mathcal{PS}_{\frac{1}{2}\alpha}(\gamma^{-1})$	Generalised Gaussian, $\mathcal{GG}(s\|\alpha)$	$\exp\left(-\|s\|^\alpha\right)$
5)	Improper Jeffrey, $\mathcal{J}(\gamma)$	γ^{-1}	Improper, $\mathcal{J}(s)$	$\|s\|^{-1}$
6)	Inverse Gaussian, $\mathcal{IG}(\gamma\|\alpha,\beta)$	$\gamma^{-\frac{3}{2}} \exp\left(-\frac{1}{2}(\alpha^2/\gamma+\beta^2\gamma)\right)$	Normal-Inv. Gaussian, $\mathcal{NIG}(s\|\alpha,\beta)$	$\mathcal{K}_1\left(\beta\sqrt{\alpha^2+s^2}\right)/\sqrt{\alpha^2+s^2}$
7)	Gamma, $\alpha,\beta > 0$, $\mathcal{G}(\gamma\|\alpha,\beta)$	$\gamma^{\alpha-1} \exp\left(-\beta\gamma\right)$	Variance Gamma, $\mathcal{VG}(s\|\alpha,\beta)$	$\|s\|^{\alpha-\frac{1}{2}} \mathcal{K}_{\alpha-\frac{1}{2}}\left(-\sqrt{\beta/2}\|s\|\right)$
8)	Dirac Mixture $\mathcal{DM}(\gamma\|\boldsymbol{\sigma}^2,\boldsymbol{\pi})$	$\sum_i \pi_i\delta(\gamma-\sigma_i^2)$	Gaussian Mixture $\mathcal{M}o\mathcal{G}(s\|\boldsymbol{\sigma}^2,\boldsymbol{\pi})$	$\sum_i \pi_i\mathcal{N}(s\|0,\sigma_i^2)$

Figure 3.1: Gaussian scale mixture potentials

1+4) Due to log-concavity for $\alpha \geq 1$, the generalised Gaussian distribution also enjoys popularity since it includes the Gaussian and the double exponential distribution [Box and Tiao, 1973, West, 1987, ch. 3.2].

2) The most common mixture is the Student's t distribution, e.g. in the relevance vector machine [Tipping, 2001].

3) One needs to combine a countably infinite amount of Gaussians to get the logistic distribution, which is closely related to the popular classification likelihood [Stefanski, 1990].

5) Ignorance w.r.t. to the scale of γ can be captured by the non-informative parameter-free but improper Jeffrey's prior [Figueiredo, 2002] as scale distribution. But, the density $\mathcal{PS}_\alpha(\gamma)$ of positive stable distributions is non-analytic. Generalised hyperbolic distributions in particular 6+7) are also used in sparse linear models [e.g. Caron and Doucet, 2008], where $\mathcal{K}_\nu(\cdot)$ is the modified Bessel function of the second kind.

8) Finally, the popular spike and slab *models corresponds to a finite Gaussian mixture with $n = 2$. The list is far from complete; α-stable distributions and symmetrised Gamma distributions are used to model images statistics [Wainwright and Simoncelli, 2000], for example.*

$$
\begin{aligned}
\ln Z &\overset{c}{=} \ln \int \mathcal{N}(\mathbf{y}|\mathbf{X}\mathbf{u}, \sigma^2\mathbf{I}) \prod_{j=1}^{q} \mathcal{T}_j(s_j) \mathrm{d}\mathbf{u}, \ \mathbf{s} = \mathbf{B}\mathbf{u} \\
&\overset{\text{(eq. 2.17)}}{=} \ln \int_{\boldsymbol{\gamma}\succeq 0} \tilde{Z}(\boldsymbol{\gamma}) \prod_{j=1}^{q} \frac{\mathbb{P}_j(\gamma_j)}{\sqrt{2\pi\gamma_j}} \mathrm{d}\boldsymbol{\gamma} \approx \ln \tilde{Z}(\boldsymbol{\gamma}^\star) \qquad (3.2) \\
&\qquad \boldsymbol{\gamma}^\star = \arg\max_{\boldsymbol{\gamma}\succeq 0} \ln \tilde{Z}(\boldsymbol{\gamma}) - \frac{1}{2}\ln|\boldsymbol{\Gamma}| + \sum_{j=1}^{q} \ln \mathbb{P}_j(\gamma_j),
\end{aligned}
$$

where the integration w.r.t. $\mathbf{u}$ and $\boldsymbol{\gamma}$ are interchanged and the scale parameters $\boldsymbol{\gamma}$ are found via MAP estimation . Instead of MAP estimation, we can apply bounding, which leads to the same variational bound as in section 3.3 auf der nächsten Seite as shown in appendix E.6 auf Seite 145. In SBL, Student's t potentials (see table 3.1), where – for a particular choice for the parameters in the Gamma scale distribution – the scalar terms $\ln \mathbb{P}_j(\gamma_j) = 0$ vanish rendering the optimisation very simple. In the process of MAP estimation for SBL (equivalent to equation 3.2)

$$
\boldsymbol{\gamma}^\star = \arg\min_{\boldsymbol{\gamma}\succeq 0} \ln|\mathbf{A}| + \ln|\boldsymbol{\Gamma}| + \frac{1}{\sigma^2} \min_{\mathbf{u}} \mathbf{u}^\top\mathbf{A}\mathbf{u} - 2\mathbf{d}^\top\mathbf{u}, \qquad (3.3)
$$

many of the values γ_j become zero, i.e. the posterior approximation collapses to a delta-distribution for some potentials.[3] Although algorithmically efficient, the degenerate posterior makes drastically overconfident uncertainty statements, which prevents successful experimental design as experienced in the image acquisition application of chapter 5. Therefore, the applications for SBL rather lie in the domain of efficient estimation rather than proper assessment

[3] The one-dimensional equivalent to equation 3.3 for $\mathbf{X} = \sigma = 1$ is $\gamma^\star = \arg\min_{\gamma\geq 0} \ln(\gamma+1) - d^2/(\gamma^{-1}+1)$ implying $\gamma^\star = \max(0, d^2-1)$. For $d \leq 1$, $\gamma^\star = 0$ the potential is pruned out and $d > 1$, $\gamma^\star > 0$ keeps the potential in the model.

of posterior uncertainty. Finally, from the theory point of view it is dangerous to maximise an approximation to the marginal likelihood since it is not clear whether the underlying exact marginal likelihood is maximised or the approximation deteriorates. In the next section, we will maximise a lower bound, which is theoretically more profound while retaining the same computational complexity as SBL.

3.3 Variational bounds

Besides the scale mixture representation, there is a variational representation of super-Gaussian potentials as a maximum over scaled Gaussians

$$\mathcal{T}(s) = \max_{\gamma>0} \mathcal{N}(s|0,\gamma)f(\gamma).$$

In particular, if $g(x) = \ln \mathcal{T}(s)$, $x = s^2$ is a decreasing and convex function of $x > 0$, then $\mathcal{T}(s)$ can be represented by a maximum over scaled Gaussians $\mathcal{T}(s) = \max_{\gamma>0} \mathcal{N}(s|0,\gamma)f(\gamma)$ and if in addition the higher-order derivatives obey $g^{(2n+1)}(x) \leq 0$, $g^{(2n)}(x) \geq 0$, then a scale mixture representation $\mathcal{T}(s) = \int_0^\infty \mathcal{N}(s|0,\gamma)\mathbb{P}(\gamma)d\gamma$ is possible [Palmer et al., 2006].

The applicability to a bigger class of super-Gaussian potentials of the variational representation comes at a cost: the parameters γ are variational parameters; they do not have a direct statistical semantic as a variance.

3.3.1 Individual potential bounds

As already described in chapter 2.5.9, we use variational lower bounds on every individual non-Gaussian potential

$$\mathcal{T}_j(s_j) \geq \exp\left(\frac{\beta_j(\gamma_j)}{\sigma^2}s_j - \frac{1}{2\sigma^2\gamma_j}s_j^2 - \frac{h_j(\gamma_j)}{2}\right) = \hat{\mathcal{T}}_j(s_j;\gamma_j) \propto \mathcal{N}\left(s_j|\beta_j\gamma_j, \sigma^2\gamma_j\right) \tag{3.4}$$

to obtain the well-known variational relaxation [Girolami, 2001, Palmer et al., 2006, Jaakkola, 1997] $\ln Z_{VB}$ of the log partition function $\ln Z$

$$\ln Z \quad \geq \quad \ln C_{\mathcal{N}}C_{\mathcal{T}} + \ln \int \mathcal{N}(\mathbf{y}|\mathbf{X}\mathbf{u},\sigma^2\mathbf{I}) \prod_{j=1}^{q} \hat{\mathcal{T}}_j(\mathbf{b}_j^\top\mathbf{u})d\mathbf{u} = \ln Z_{VB}(\gamma) = \ln \tilde{Z}(\gamma) - \frac{1}{2}\sum_{j=1}^{q} h(\gamma_j).$$

In the following, we drop the index j to increase clarity and focus on symmetric (even) potentials $\mathcal{T}(s) = \mathcal{T}(-s)$ with symmetric lower bounds $\hat{\mathcal{T}}(s;\gamma) = e^{-s^2/(2\sigma^2\gamma)-h(\gamma)/2}$. However, lower bounds can also be obtained for non-symmetric potentials, too:
first, the cumulative logistic potential (see figures 2.2a and 3.2) can be symmetrised, i.e. $e^{-\beta s}\mathcal{T}(s)$ is symmetric with (constant in γ) $\beta = \frac{c}{2}$, c being the class label.
Second, shifting and scaling of s, and scaling of the potential itself can be easily achieved be modifying the bound

$$\mathcal{T}(s) \geq \hat{\mathcal{T}}(s;\gamma) \Rightarrow a \cdot \mathcal{T}\left(\frac{s-d}{g^2}\right) \geq a \cdot \hat{\mathcal{T}}\left(\tilde{s};\tilde{\gamma}\right), \, \tilde{s} = s - d, \, \tilde{\gamma} = g^2\gamma.$$

The analytical expression for the bounds are obtained by exploiting the (strong) super-Gaussianity of the potential $\mathcal{T}(s)$. Strong super-Gaussianity implies that $g(s) = \ln \mathcal{T}(s)$ is convex and decreasing as a function of $x = s^2/\sigma^2$ [Palmer et al., 2006]. We write $g(x)$ in the sequel. Fenchel duality [Rockafellar, 1970, section 12], allows to represent $g(x)$ in a variational form using the conjugate function $g^*(p)$

$$g(x) = \max_p xp - g^*(p) = \max_{\gamma>0} -\frac{x}{2\gamma} - g^*\left(-\frac{1}{2\gamma}\right) = \max_{\gamma>0} -\frac{x}{2\gamma} - \frac{h(\gamma)}{2},$$

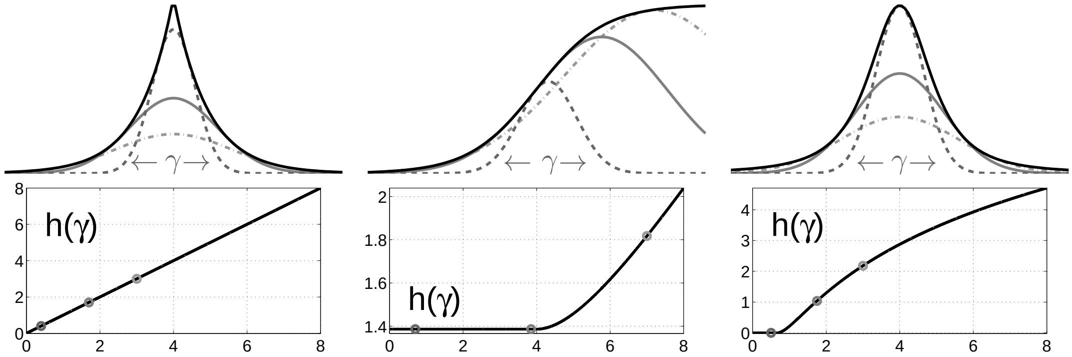


Figure 3.2: Individual potential bounds
Super-Gaussian potentials can be bounded by scaled Gaussian lower bounds of any width γ. From left to right: Laplace, cumulative logistic and Student's t distribution.

where $p = -\frac{1}{2\gamma}$ and $h(\gamma) = 2 \cdot g^*(p)$. This translates into a lower potential bound

$$\mathcal{T}(s) = \max_{\gamma>0} \exp\left(-\frac{1}{2\sigma^2\gamma}s^2 - \frac{h(\gamma)}{2}\right),\; h(\gamma) = \max_{x\geq 0} -\frac{x}{\gamma} - 2 \cdot \ln \mathcal{T}(x),$$

which is illustrated for some often used potentials in figure 3.2.

Many potentials (besides the ones in figure 3.2) are in fact super-Gaussian. All Gaussian scale mixtures $\mathcal{T}(s) = \int \mathcal{N}(s|0,\sigma^2\gamma)\mathbb{P}(\gamma)\mathrm{d}\gamma$ (figure 3.1) are super-Gaussian and the respective height function $h(\gamma)$ can be represented using $\mathbb{P}(\gamma)$ [Palmer et al., 2006]. Furthermore, mixtures of super-Gaussian potentials $\sum_i \alpha_i \mathcal{T}(\xi_i s)$, $\xi_i, \alpha_i > 0$ are super-Gaussian because the *logsumexp* function $\mathbf{x} \mapsto \ln(\mathbf{1}^\top e^{\mathbf{x}})$ is strictly convex and increasing in all x_i [Boyd and Vandenberghe, 2004, section 3.1.5].

3.3.2 Joint variational lower bound

Plugging the individual lower bounds $\mathcal{T}_j(s_j) \geq \hat{\mathcal{T}}_j(s_j;\gamma_j)$ into the log partition function

$$\ln \mathbb{P}(\mathcal{D}) = \ln Z = \ln C_{\mathcal{N}} C_{\mathcal{T}} + \ln \int \mathcal{N}(\mathbf{y}|\mathbf{X}\mathbf{u},\sigma^2\mathbf{I}) \prod_{j=1}^{q} \mathcal{T}_j(s_j)\mathrm{d}\mathbf{u}$$

and dropping all terms constant in the variational parameters γ yields the variational criterion $\phi(\gamma) \overset{c}{=} -2\ln Z_{VB}(\gamma)$ to be minimised (equation 2.20 in chapter 2.5.9)

$$\begin{aligned} \phi(\gamma) &= \overbrace{\sum_{j=1}^{q} h_j(\gamma_j)}^{h(\gamma)} + \frac{1}{\sigma^2}\min_{\mathbf{u}} \overbrace{\mathbf{u}^\top\mathbf{A}\mathbf{u} - 2\mathbf{d}^\top\mathbf{u}}^{R(\mathbf{u},\gamma)} + \ln|\mathbf{A}|, \text{ where} \\ & \mathbf{d} = \mathbf{X}^\top\mathbf{y} + \mathbf{B}^\top\boldsymbol{\beta}, \text{ and } \mathbf{A} = \mathbf{X}^\top\mathbf{X} + \mathbf{B}^\top\boldsymbol{\Gamma}^{-1}\mathbf{B}. \end{aligned} \tag{3.5}$$

For a particular value of the variational parameters γ, the posterior approximation $\mathbb{Q}(\mathbf{u}) = \mathcal{N}(\mathbf{u}|\mathbf{m},\mathbf{V})$ has mean $\mathbf{m} = \mathbf{A}^{-1}\mathbf{d} = \arg\min_{\mathbf{u}} R(\mathbf{u},\gamma)$ and variance $\mathbf{V} = \sigma^2\mathbf{A}^{-1}$ (see appendix C.1). The next section studies convexity properties of $\phi(\gamma)$. Once these are established, we will discuss efficient scalable and generic minimisation algorithms for solving $\gamma^\star = \arg\min_\gamma \phi(\gamma)$.

3.4 Convexity properties of variational inference

The basic convexity result is simple: $\phi(\gamma)$ is convex iff all strongly super-Gaussian potentials $\mathcal{T}_j(s_j)$ are log-concave. We will look at each of the three terms $\ln|\mathbf{A}|$, $R(\mathbf{u},\gamma)$, $h(\gamma)$ of (equation 3.6) in turn. We start with the log determinant, continue with the least-square term and finish with the height functions.

3.4.1 Convexity of log determinant term

Theorem 1 *Let* $\mathbf{X} \in \mathbb{R}^{m\times n}$ *and* $\mathbf{B} \in \mathbb{R}^{q\times n}$ *be arbitrary matrices and* $\mathbf{A}_{\mathbf{f}(\gamma)} = \mathbf{X}^\top\mathbf{X} + \mathbf{B}^\top dg\left(\mathbf{f}(\gamma)\right)\mathbf{B}$ *with twice continuously differentiable* $f_j(\gamma_j) > 0$ *so that* $\gamma \mapsto \ln|\mathbf{A}_{\mathbf{f}(\gamma)}|$ *exists.*

1. *If* $f_j : \mathbb{R} \to \mathbb{R}_+$ *are log-convex then* $\gamma \mapsto \ln\left|\mathbf{A}_{\mathbf{f}(\gamma)}\right|$ *is convex. For* $f_j(\gamma_j) = \gamma_j^{-1}$ *in particular,* $\gamma \mapsto \ln\left|\mathbf{A}_{\gamma^{-1}}\right|$ *is convex.*
2. *If* $f_j : \mathbb{R} \to \mathbb{R}_+$ *are concave then* $\gamma^{-1} \mapsto \ln\left|\mathbf{A}_{\mathbf{f}(\gamma^{-1})}\right|$ *is concave. For* $f_j(\gamma_j) = \gamma_j^{-1}$ *in particular,* $\gamma^{-1} \mapsto \ln\left|\mathbf{A}_{\gamma^{-1}}\right|$ *is concave.*
3. *If* $f_j : \mathbb{R} \to \mathbb{R}_+$ *are concave then* $\gamma \mapsto \mathbf{1}^\top \ln\mathbf{f}(\gamma) + \ln\left|\mathbf{A}_{[\mathbf{f}(\gamma)]^{-1}}\right|$ *is concave. For* $f_j(\gamma_j) = \gamma_j^{-1}$ *in particular,* $\gamma \mapsto \ln|\mathbf{\Gamma}| + \ln\left|\mathbf{A}_{\gamma^{-1}}\right|$ *is concave.*
4. *Let* $\mathbf{V} = \sigma^2\mathbf{A}_{\gamma^{-1}}^{-1}$ *be the posterior covariance and* $\boldsymbol{\nu} = dg(\mathbf{B}\mathbf{V}\mathbf{B}^\top) = \mathbb{V}_{\mathrm{Q}}[\mathbf{s}|\mathcal{D}]$ *the marginal variances of* $\mathbf{s} = \mathbf{B}\mathbf{u}$. *Then, we can bound the marginal variances by* $\mathbf{0} \preceq \boldsymbol{\nu} \preceq \sigma^2\gamma$.

Part (1) that is novel to our knowledge is proven in appendix E.1, part (2) is obtained by combining classical results about convex functions [Boyd and Vandenberghe, 2004, sections 3.1.5/3.2.4] and having in mind that $\gamma^{-1} \mapsto \ln|\mathbf{A}_{\gamma^{-1}}|$ is nondecreasing in every component γ_j^{-1}. Part (3) is proven in appendix E.2 and the upper bound in part (4) can be seen componentwise from

$$\begin{aligned}\nu_j &= \sigma^2\mathbf{b}_j^\top\mathbf{A}_{\gamma^{-1}}^{-1}\mathbf{b}_j = \sigma^2\max_{\mathbf{u}} 2\mathbf{b}_j^\top\mathbf{u} - \mathbf{u}^\top(\mathbf{X}^\top\mathbf{X} + \mathbf{B}^\top\mathbf{\Gamma}^{-1}\mathbf{B})\mathbf{u}\\ &\leq \sigma^2\max_{\mathbf{u}} 2\mathbf{b}_j^\top\mathbf{u} - \mathbf{s}^\top\mathbf{\Gamma}^{-1}\mathbf{s} \leq \sigma^2\max_{s_j} 2s_j - s_j^2\gamma_j^{-1} = \sigma^2\gamma_j,\ \mathbf{s} = \mathbf{B}\mathbf{u}.\end{aligned}$$

Thus, the term $\ln|\mathbf{A}|$ is in the variational criterion (equation 3.6) is convex in γ.

3.4.2 Convexity of least-square term

The term $R(\mathbf{u},\gamma) = \mathbf{u}^\top\mathbf{A}\mathbf{u} - 2\mathbf{d}^\top\mathbf{u} = \|\mathbf{X}\mathbf{u} - \mathbf{y}\|^2 + \mathbf{s}^\top\mathbf{\Gamma}^{-1}\mathbf{s} - 2\boldsymbol{\beta}^\top\mathbf{s}$ is jointly convex in $(\mathbf{u},\gamma)$ since it is a sum of jointly convex terms: $\|\mathbf{X}\mathbf{u} - \mathbf{y}\|^2 - 2\boldsymbol{\beta}^\top\mathbf{s}$ is a positive semi-definite quadratic in $\mathbf{u}$ and $\mathbf{s}^\top\mathbf{\Gamma}^{-1}\mathbf{s}$ is a *quadratic−over−linear* function in $\mathbf{u},\gamma$, which is convex [Boyd and Vandenberghe, 2004, chapter 3.1.5].

Furthermore, minima of jointly convex functions w.r.t. some of the arguments yield convex functions [Boyd and Vandenberghe, 2004, chapter 3.2.5] implying convexity of $\gamma \mapsto \min_{\mathbf{u}} R(\mathbf{u},\gamma)$.

3.4.3 Convexity of height functions

In appendix E.3, we show that for strongly super-Gaussian potentials (e.g. Gaussian scale mixtures) convexity of $h_j(\gamma_j)$ is equivalent to log-concavity of the the potential $\mathcal{T}_j(s_j)$. Therefore, $h(\gamma) = \sum_{j=1}^q h_j(\gamma_j)$ is a convex function whenever all potentials are log-concave. The respective expressions for the bounds shown in figure 3.2 are summarised in table 3.1.

3.4.4 Summary

Theorem 2 *Let* $\mathbf{X} \in \mathbb{R}^{m\times n}$ *and* $\mathbf{B} \in \mathbb{R}^{q\times n}$ *be arbitrary matrices and let* $\mathbb{P}(\mathbf{u}|\mathbf{y})$ *be the posterior of a model with strongly super-Gaussian potentials* $\mathcal{T}_j(s_j)$ *of the form* $\mathbb{P}(\mathbf{u}|\mathbf{y}) \propto \mathcal{N}(\mathbf{y}|\mathbf{X}\mathbf{u},\sigma^2\mathbf{I})\prod_{j=1}^q\mathcal{T}_j(s_j)$ *with* $\mathbf{s} = \mathbf{B}\mathbf{u}$. *Further let* $\phi(\gamma) = h(\gamma) + \frac{1}{\sigma^2}\min_{\mathbf{u}} R(\mathbf{u},\gamma) + \ln|\mathbf{A}|$ *be the variational criterion from equation 3.6 for the individual potential bound relaxation* $\ln Z \overset{c}{\geq} -\frac{1}{2}\phi(\gamma)$.

Potential	$\mathcal{T}(s) =$	$h(\gamma) =$
Laplace	$\exp(-\tau\lvert s\rvert)$	$\tau^2\gamma$
Student's t	$(1+\frac{\tau}{\nu}s^2)^{-\frac{\nu+1}{2}}$	$\begin{cases} 0 & \gamma \leq \frac{\nu}{\tau(\nu+1)} \\ (\nu+1)\ln\left(\gamma\tau\frac{\nu+1}{\nu}\right) - (\nu+1) + \frac{\nu}{\tau\gamma} & \gamma > \frac{\nu}{\tau(\nu+1)} \end{cases}$
Logistic	$[\cosh(\tau s)]^{-2}$	$\begin{cases} 0 & \gamma \leq \frac{1}{2\tau^2} \\ 4\ln\cosh(g_\gamma) - 2g_\gamma\tanh(g_\gamma) & \gamma > \frac{1}{2\tau^2} \end{cases}$
Cumulative logistic	$\exp\left(\frac{cs}{2}\right)[2\cosh(cs)]^{-1}$	$2\ln 2 + \begin{cases} 0 & \gamma \leq 4 \\ 2\ln\cosh(g_\gamma) - g_\gamma\tanh(g_\gamma) & \gamma > 4 \end{cases}$

Table 3.1: Height functions for individual potential bounds
For the logistic and the cumulative logistic potential, we used the function $g_\gamma = g(\gamma) = f^{-1}(\gamma)$ defined as the inverse function of $f(x) = x\coth(x)$. In fact, the cumulative logistic height function $h_{CL}(\gamma)$ can be written as $h_{CL}(\gamma) = \ln 2 + \frac{1}{2}h_L(\gamma)$, where $h_L(\gamma)$ is the height of the logistic potential and $\tau = \sqrt{2}$. We use binary class labels $c \in \{\pm 1\}$.

1. *If all potentials $\mathcal{T}_j(s_j)$ are log-concave then $\phi(\gamma)$ is convex and is one potential $\mathcal{T}_j(s_j)$ is not log-concave, one can find $\mathbf{X}$, $\mathbf{B}$ and $\mathbf{y}$ so that $\phi(\gamma)$ is not convex.*

Note that the Gaussian log partition function $\ln \tilde{Z}(\gamma) = \int \mathcal{N}(\mathbf{y}|\mathbf{X}\mathbf{u}, \sigma^2\mathbf{I}) \prod_{j=1}^{q} \tilde{\mathcal{T}}_j(s_j) d\mathbf{u}$, where the non-Gaussian potentials $\mathcal{T}_j(s)$ have been replaced by Gaussians $\tilde{\mathcal{T}}_j(s_j) = \exp(\frac{\beta_j}{\sigma^2}s_j - \frac{1}{2\sigma^2\gamma_j}s_j^2)$ can be written as $\ln \tilde{Z}(\gamma) \overset{c}{=} -\frac{1}{2}[\phi(\gamma) - h(\gamma)]$. It is well known, that $\gamma^{-1} \mapsto \ln \tilde{Z}$ is convex, i.e. $\gamma^{-1} \mapsto \phi(\gamma) - h(\gamma)$ is concave since γ^{-1} are the natural parameters of an exponential family graphical model [Wainwright and Jordan, 2008]. However, the convexity of $\gamma \mapsto \ln \tilde{Z}$ did not receive attention so far and seems to be a special property of the Gaussian case. However, the knowledge that $\gamma^{-1} \mapsto h(\gamma)$ is convex for any strongly super-Gaussian potential, does not reveal any new insights about the concavity properties of $\gamma^{-1} \mapsto \phi(\gamma)$.

Our result settles a longstanding problem in approximate inference: if the posterior mode of a super-Gaussian model can be found via a convex problem, then a frequently used approximation [Girolami, 2001, Palmer et al., 2006, Jaakkola, 1997] is convex as well.

Convexity of the objective $\phi(\gamma)$ is highly desirable for several reasons: there are no local minima problems, i.e. no cumbersome restarting is needed in the optimisation algorithm. Furthermore, the results are typically robust to small perturbation of the input. However, convexity of $\phi(\gamma)$ alone does not lead to an efficient minimisation algorithm. In the next section, we will propose a class of algorithms solving the variational problem $\phi(\gamma)$ efficiently in high dimensions by decoupling the criterion.

3.5 Scalable optimisation algorithms

We start by restating the variational inference objective $\phi(\gamma)$ from equation 3.5

$$\phi(\gamma, \mathbf{u}) = h(\gamma) + \frac{1}{\sigma^2}R(\mathbf{u}, \gamma) + \ln|\mathbf{A}|, \ \mathbf{A} = \mathbf{X}^\top\mathbf{X} + \mathbf{B}^\top\mathbf{\Gamma}^{-1}\mathbf{B}, \ \phi(\gamma) = \min_{\mathbf{u}} \phi(\gamma, \mathbf{u}), \tag{3.6}$$

where $\mathbf{s} = \mathbf{B}\mathbf{u}$ and $R(\mathbf{u}, \gamma) = \|\mathbf{X}\mathbf{u} - \mathbf{y}\|^2 + \mathbf{s}^\top\mathbf{\Gamma}^{-1}\mathbf{s} - 2\boldsymbol{\beta}^\top\mathbf{s}$. We know that $\phi(\gamma)$ is convex whenever all potentials are log-concave. The general wisdom in mathematical programming is that convex optimisation is well understood and basically a solved problem; the division line being in optimisation is between convex and non-convex optimisation [Boyd and Vandenberghe, 2004] rather than between linear and non-linear optimisation. For our special case, however, we additionally require computational efficiency and hence scalability.

Already a single exact gradient computation

$$\frac{\partial\phi(\gamma, \mathbf{u})}{\partial\gamma} = \sum_{j=1}^{q} h'_j(\gamma_j) - \gamma^{-2}\left[\frac{1}{\sigma^2}\mathbf{s}\odot\mathbf{s} + \mathrm{dg}(\mathbf{B}^\top\mathbf{A}^{-1}\mathbf{B})\right]$$

is very costly for models with large numbers of variables n because matrix inversion in the $\mathrm{dg}(\mathbf{B}^\top\mathbf{A}^{-1}\mathbf{B})$ term is an $\mathcal{O}(n^3)$ operation that cannot be circumvented. Therefore, standard joint optimisation code like conjugate gradients (CG) or successful quasi-Newton methods such as BFGS do not scale well with the size of the model n even if $\mathbf{B}$ and $\mathbf{X}$ are highly structured matrices.

Another line of attack is coordinate descent, that is iterating over the potentials $j = 1..q$ while optimising with respect to a single γ_j at a time as done in Girolami [2001]. Making use of appendices A.1.1 and A.1.2, the objective restricted to γ_j is given by

$$\begin{aligned}\phi_j(\gamma_j) &= h_j(\gamma_j) - \frac{1}{\sigma^2}\mathbf{d}^\top\left(\mathbf{A}_{\neg j} + \mathbf{b}_j\gamma_j^{-1}\mathbf{b}_j^\top\right)^{-1}\mathbf{d} + \ln\left|\mathbf{A}_{\neg j} + \mathbf{b}_j\gamma_j^{-1}\mathbf{b}_j^\top\right|,\ \mathbf{d} = \mathbf{X}^\top\mathbf{y} + \mathbf{B}^\top\boldsymbol{\beta} \\ &\overset{c}{=} h_j(\gamma_j) + \frac{(\mathbf{d}^\top\mathbf{v}_j/\sigma)^2}{\gamma_j + \mathbf{b}_j^\top\mathbf{v}_j} + \ln(1+\gamma_j^{-1}\mathbf{b}_j^\top\mathbf{v}_j),\ \mathbf{v}_j = \mathbf{A}_{\neg j}^{-1}\mathbf{b}_j.\end{aligned}$$

As a result, we can optimise $\phi(\gamma)$ w.r.t. γ_j by solving the linear system[4] $\mathbf{v}_j = \mathbf{A}_{\neg j}^{-1}\mathbf{b}_j$ of size $n \times n$ and using standard convex optimisation techniques in 1-d. Again, if q and n are large, such algorithms are intractable even for highly structured matrices.

We therefore need an approach satisfied with a small number of these expensive calculations and exploits structure of $\phi(\gamma)$ other than its convexity. Our double loop algorithm as proposed in the following, decouples the objective and minimises a simple surrogate function that is iteratively updated instead. Consequently, we need to solve only few linear systems to minimise $\phi(\gamma)$.

3.5.1 Facts about the objective function

Let us collect some facts about the optimisation problem $\min_\gamma \phi(\gamma)$ of equation 3.6, which go beyond joint convexity properties of $\phi(\gamma, \mathbf{u})$ as proven in section 3.4. First of all, the terms $R(\mathbf{u}, \gamma)$ and $\ln|\mathbf{A}|$ are jointly convex independently of the type of potentials as proven in section 3.4; only $h(\gamma)$ depends on the potentials itself.

1. Joint convexity allows to interchange the order of minimisation between the variables $\min_\gamma \min_{\mathbf{u}} \phi(\gamma, \mathbf{u}) = \min_{\mathbf{u}} \min_\gamma \phi(\gamma, \mathbf{u})$.

2. Fixing γ, the criterion $\phi(\gamma, \mathbf{u})$ is a quadratic function in $\mathbf{u}$ amenable to efficient and scalable minimisation schemes such as conjugate gradients (CG) or iteratively reweighted least squares (IRLS) as described in section 2.3.1.

3. The terms $h(\gamma)$ and $R(\mathbf{u}, \gamma)$ naturally decouple or decompose into a sum over the single components γ_j since

$$h(\gamma) + \frac{1}{\sigma^2}R(\mathbf{u}, \gamma) \overset{c}{=} \sum_{j=1}^{q}\left[h_j(\gamma_j) + \frac{s_j^2}{\sigma^2\gamma_j}\right], \tag{3.7}$$

   where we dropped terms not depending on γ. Decoupling in γ is highly desirable since it reduces a q-dimensional minimisation to q simple 1-dimensional minimisations.

4. The following facts are known about the coupled term $\ln|\mathbf{A}|$: the function $\gamma \mapsto \ln|\mathbf{A}|$ is convex whereas $\gamma^{-1} \mapsto \ln|\mathbf{A}|$, $\gamma \mapsto \ln|\mathbf{\Gamma}|$ and $\gamma \mapsto \ln|\mathbf{A}| + \ln|\mathbf{\Gamma}| = \ln|\mathbf{A}\mathbf{\Gamma}|$ are concave.

We will exploit the facts 1-4 in various ways to construct efficient minimisation schemes.

3.5.2 Double loop minimisation

A powerful class of ideas dealing with non-convex minimisation problems are so-called *double loop* algorithms, also known as convex-concave programming (CCCP) or difference of convex

[4] Solving a linear system with conjugate gradients is a scalable operation as long as the matrix-vector-multiplication with the system matrix $\mathbf{A}$ is faster than $\mathcal{O}(n^2)$.

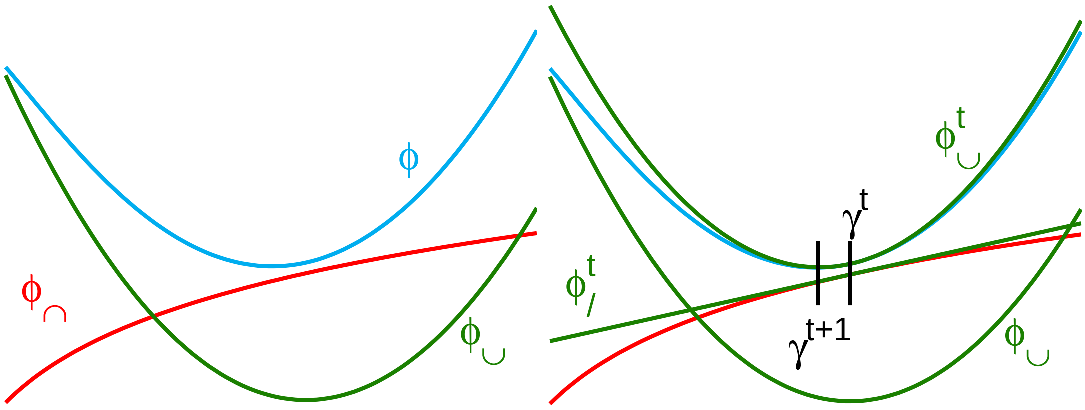


Figure 3.3: Double loop algorithm
Minimisation of a non-convex objective $\phi(\gamma) = \phi_\cap(\gamma) + \phi_\cup(\gamma)$ by linearly upper bounding the concave part $\phi_\cap(\gamma) \le \phi^t_/(\gamma)$ and minimising the surrogate function $\phi^t(\gamma) = \phi^t_/(\gamma) + \phi_\cup(\gamma)$ instead. If we iterate over t, the algorithm will converge to a point with $\frac{\partial \phi}{\partial \gamma} = \mathbf{0}$.

(DC) programming approaches. In statistics, machine learning and computer vision, these algorithms are widespread: the expectation-maximisation method [Dempster et al., 1977], CCCP [Yuille and Rangarajan, 2003] for approximate inference in discrete models or variational mean field [Attias, 2000] for continuous models being only among the most prominent examples.

The basic underlying idea is the decomposition of the objective function $\phi(\gamma) = \phi_\cap(\gamma) + \phi_\cup(\gamma)$ into a convex part $\phi_\cup(\gamma)$ and a concave part $\phi_\cap(\gamma)$, which is possible for any function. In every iteration t of the algorithm, the concave part is upper bounded by a linear function $\phi_\cap(\gamma) \le \phi^t_/(\gamma)$ tight at the current location γ^t and the (hence convex) surrogate function $\phi^t(\gamma) = \phi^t_/(\gamma) + \phi_\cup(\gamma)$ is minimised to yield the next location $\gamma^{t+1} = \arg\min_\gamma \phi^t(\gamma)$ as illustrated in figure 3.3 and detailed in algorithm 3.1. Under mild conditions, the sequence $\{\gamma^t\}_{t=1..T}$ converges to a stationary point of the exact criterion $\phi(\gamma)$. Refitting the bound $\phi^t_/(\gamma)$ or iterating over t is referred to as the *outer loop* and minimising the surrogate function $\phi^t(\gamma)$ is termed the *inner loop*. Since the upper bounds $\phi^t_/(\gamma) \overset{c}{=} \mathbf{z}_1^\top \gamma$ and $\phi^t_/(\gamma^{-1}) \overset{c}{=} \mathbf{z}_2^\top \gamma^{-1}$ have to be tight at the current location γ^t, their respective slopes $\mathbf{z}_{1,2}$ are given by $\mathbf{z}_1 = \frac{\partial}{\partial \gamma}\phi(\gamma^t)$ and $\mathbf{z}_2 = -\gamma^2 \odot \frac{\partial}{\partial \gamma}\phi(\gamma^t)$.

We use the double loop ideas not only to deal with non-log-concave potentials such as the Student's t potential, where the height function $h_j(\gamma_j)$ is not convex, but most importantly we use double loop algorithms to decouple the $\ln|\mathbf{A}|$ part of $\phi(\gamma)$ by a linear upper bound. From Fenchel and using fact 4 from section 3.5.1 duality there are two possible bounds:

$$\begin{array}{rrcrcrcrcl} (1) & \phi_\cap(\gamma) & = & \ln|\mathbf{A}| + \ln|\mathbf{\Gamma}| & \le & \mathbf{z}^\top\gamma - \phi_\cap^*(\mathbf{z}) & \overset{c}{=} & \sum_{j=1}^q z_j\gamma_j & =: & \phi_/(\gamma), \text{ and} \\ (2) & \phi_\cap(\gamma^{-1}) & = & \ln|\mathbf{A}| & \le & \mathbf{z}^\top\gamma^{-1} - \phi_\cap^*(\mathbf{z}) & \overset{c}{=} & \sum_{j=1}^q z_j\gamma_j^{-1} & =: & \phi_/(\gamma^{-1}). \end{array} \tag{3.8}$$

Figure 3.4 provides a graphical illustration. As a result, we can upper bound $\ln|\mathbf{A}|$ itself by the two convex expressions $\phi^{(1)}_\cup(\gamma)$ and $\phi^{(2)}_\cup(\gamma)$

$$\mathbf{z}^\top_{\ln|\mathbf{A\Gamma}|}\gamma - \mathbf{1}^\top \ln\gamma \overset{c}{=} \phi^{(1)}_\cup(\gamma) \ge \ln|\mathbf{A}| \le \phi^{(2)}_\cup(\gamma) \overset{c}{=} \mathbf{z}^\top_{\ln|\mathbf{A}|}\gamma^{-1}, \tag{3.9}$$

where we dropped the offsets independent of γ. We can see from figure 3.4 that $\phi^{(1)}_\cup(\gamma)$ reflects the behaviour of $\ln|\mathbf{A}|$ more faithfully for large values of γ and overestimates $\ln|\mathbf{A}|$ for small γ. In turn, $\phi^{(2)}_\cup(\gamma)$ is relatively exact for small γ but rather loose for large γ. During the optimisation, $\phi^{(1)}_\cup(\gamma)$ favours larger γ and $\phi^{(2)}_\cup(\gamma)$ prefers smaller γ. While double loop algorithms have been proposed for non-convex approximate inference, we show that they can also be used

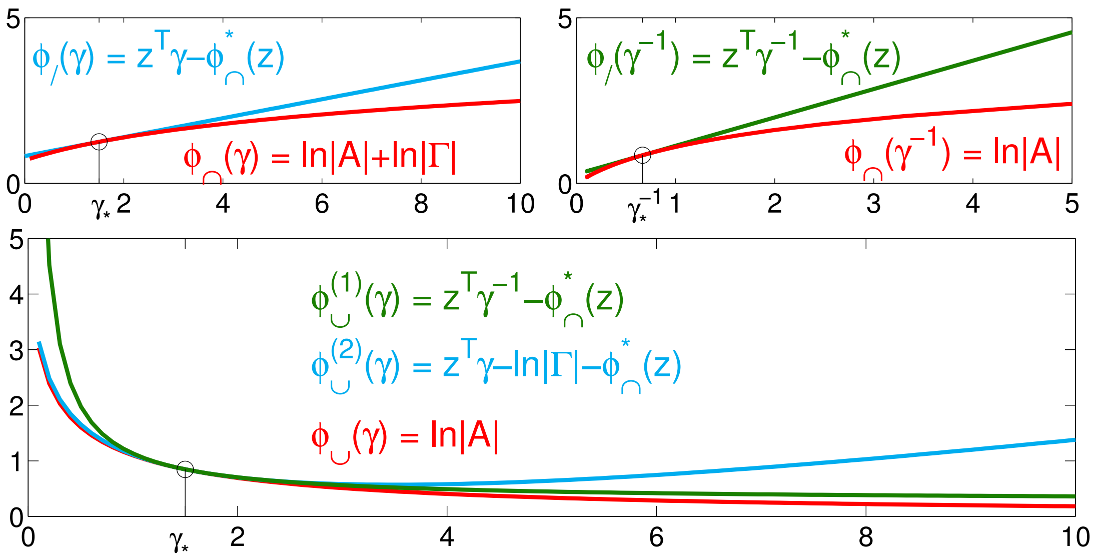


Figure 3.4: Two log determinant bounds

Two ways of upper bounding concave functions containing $\ln|\mathbf{A}| = \ln|\mathbf{X}^\top\mathbf{X} + \mathbf{B}^\top\mathbf{\Gamma}^{-1}\mathbf{B}|$ *by linear functions in order to decouple them. Note that the upper left bound is linear in* γ *whereas the upper right bound is linear in* γ^{-1}*. As shown in the lower plot, both upper bounds are however convex in* γ *and decompose into a sum. The example uses* $m = n = q = 1$ *and* $\mathbf{X} = 1$, $\mathbf{B} = \sqrt{2}$*; the bounds are tight at* $\gamma_* = 1.5$*. Note that the two bounds are tighter for different values of* γ*.*

to drastically speed up the optimisation of *convex* inference problems. No matter, which bound is used in practise, the resulting algorithm is globally convergent.

3.5.3 Practical decompositions

In case, some potentials $\mathcal{T}_j(s_j)$ are not log-concave, we can decompose the height functions into a convex and a concave part $h(\gamma) = h_\cap(\gamma) + h_\cup(\gamma)$; if all $\mathcal{T}_j(s_j)$ are log-concave, then $h_\cap(\gamma) \equiv 0$. We can use the same bounding idea to obtain $h(\gamma) \le \mathbf{z}_{h_\cap}^\top \gamma - h_\cap^*(\mathbf{z}) + h_\cup(\gamma) \overset{c}{=} \mathbf{z}_{h_\cap}^\top \gamma + h_\cup(\gamma)$. In combination with the two possibilities of equation 3.9 to decouple $\ln|\mathbf{A}|$, we get the general decomposition

$$\begin{aligned}\phi(\gamma) \;&\overset{c}{\le}\; \min_{\mathbf{u}} h_\cup(\gamma) + \frac{1}{\sigma^2}R(\mathbf{u},\gamma) + (\overbrace{\mathbf{z}_{h_\cap} + \mathbf{z}_{\ln|\mathbf{A}\Gamma|}}^{\mathbf{z}_1 \succeq \mathbf{0}})^\top \gamma + (\overbrace{\mathbf{z}_{\ln|\mathbf{A}|}}^{\mathbf{z}_2 \succeq \mathbf{0}})^\top \gamma^{-1} - (\overbrace{\operatorname{sign}(\mathbf{z}_{\ln|\mathbf{A}\Gamma|})}^{\mathbf{z}_3 \in \{0,1\}^q})^\top \ln\gamma \\ &=: \min_{\mathbf{u}} \phi_{\mathbf{z}}(\gamma, \mathbf{u}) \qquad (3.10)\end{aligned}$$

where $\mathbf{z}_1$ contains the sum of the weights for the bounds on $h_\cap(\gamma)$ and $\ln|\mathbf{A}| + \ln|\mathbf{\Gamma}|$, respectively. The presence of $\mathbf{z}_{\ln|\mathbf{A}\Gamma|} \succ \mathbf{0}$ switches on the respective, components of the indicator vector $\mathbf{z}_3 \in \{0,1\}^q$. Furthermore, $\mathbf{z}_2 \succeq \mathbf{0}$ is the weight for the bound on $\ln|\mathbf{A}|$. For convex $h(\gamma)$, we have $\mathbf{z}_{h_\cap} = \mathbf{0}$.

In theory, both types of bounds $\phi_\cup^{(1)}(\gamma)$ and $\phi_\cup^{(2)}(\gamma)$ can be used; also convex combinations $\alpha\phi_\cup^{(1)}(\gamma) + (1-\alpha)\phi_\cup^{(2)}(\gamma)$, $\alpha \in [0,1]$ can be used without any additional computational effort. In our implementation (see section 3.6) and experiments, we use the direct approach via $\phi_\cup^{(2)}(\gamma)$, where $\mathbf{z}_1 = \mathbf{z}_3 = \mathbf{0}$. For the non-log-concave Student's t potential (see table 3.1), where $h_\cup(\gamma) = \frac{\nu}{\tau}\gamma^{-1}$ and $h_\cap(\gamma) = (\nu+1)\ln\gamma \neq 0$, we naturally obtain $\mathbf{z}_{h_\cap} \succeq \mathbf{0}$ suggesting the $\phi_\cup^{(1)}(\gamma)$ bound.

Using fact 1 from section 3.5.1 and joint convexity of the surrogate objective $\phi_{\mathbf{z}}(\mathbf{u},\gamma)$, we can interchange the order of minimisation $\min_\gamma \min_{\mathbf{u}} \phi_{\mathbf{z}}(\mathbf{u},\gamma) = \min_{\mathbf{u}} \min_\gamma \phi_{\mathbf{z}}(\mathbf{u},\gamma)$. Combined

***Algorithm 3.1** General double loop variational inference algorithm*

```
$\boxed{\textbf{Outer loop:}}$ marginal variances $\boldsymbol{\nu} = \mathrm{dg}\left(\mathbf{B}\mathbb{V}_{Q(\mathbf{u}|\mathcal{D})}[\mathbf{u}]\mathbf{B}^\top\right)$ by Lanczos (section 3.5.4)
Refit upper bound $\phi_{\mathbf{z}}(\boldsymbol{\gamma}, \mathbf{u})$ of equation 3.10
repeat
  if $\phi_{\cup}^{(1)}$ bound used then
    $\mathbf{z}_1 \leftarrow \mathbf{z}_{h_\cap} - \boldsymbol{\gamma}^2 \odot \boldsymbol{\nu}/\sigma^2 + \boldsymbol{\gamma}^{-1}$, $\mathbf{z}_2 \leftarrow \mathbf{0}$
  else
    $\mathbf{z}_1 \leftarrow \mathbf{z}_{h_\cap}$, $\mathbf{z}_2 \leftarrow \boldsymbol{\nu}/\sigma^2$
  end if
  $\boxed{\textbf{Inner loop:}}$ marginal means $\mathbf{u}_* = \mathbb{E}_{Q(\mathbf{u}|\mathcal{D})}[\mathbf{u}]$ by IRLS (section 3.5.5)
  if First outer loop then
    Init $\mathbf{u} \leftarrow \mathbf{0}$.
  else
    Initialise $\mathbf{u} \leftarrow \mathbf{u}_*$ (previous solution).
  end if
  Find $\mathbf{u}_* \leftarrow \arg\min_{\mathbf{u}} \phi_{\mathbf{z}}(\mathbf{u})$ of equation 3.12
  repeat
    Solve linear system $\frac{\partial^2 \phi_{\mathbf{z}}(\mathbf{u})}{\partial \mathbf{u} \partial \mathbf{u}^\top}\mathbf{d} \leftarrow -\frac{\partial \phi_{\mathbf{z}}(\mathbf{u})}{\partial \mathbf{u}}$ by CG to obtain Newton direction $\mathbf{d}$
    Find step size $\lambda$ by line search along $\phi_{\mathbf{z}}(\mathbf{u} + \lambda \mathbf{d})$
    Update $\mathbf{u} \leftarrow \mathbf{u} + \lambda \mathbf{d}$
  until Inner loop converged
  Update $\mathbf{s} = \mathbf{B}\mathbf{u}_*$, $\gamma_j \leftarrow \arg\min_\gamma h_j(s_j, \gamma_j)$ of equation 3.11
until Outer loop converged
```

The objective $\phi(\boldsymbol{\gamma}, \mathbf{u})$ of equation 3.6 is jointly minimised w.r.t. $\boldsymbol{\gamma}$ and $\mathbf{u}$ by refitting an auxiliary upper bound $\phi_{\mathbf{z}}(\boldsymbol{\gamma}, \mathbf{u})$ in every outer loop iteration, which is then minimised in the inner loop by a Newton algorithm. Both the inner and the outer loop use standard computational linear algebra tools like conjugate gradients and Lanczos as numerical primitives. All computations are reduced to matrix vector multiplications with $\mathbf{B}$ and $\mathbf{X}$ rendering the approach scalable.

with the decoupling in $\boldsymbol{\gamma}$ (section 3.5.1 fact 3) and the definition

$$h_j^*(s_j) = \frac{\sigma^2}{2}\min_{\gamma_j} h_j(s_j, \gamma_j), \quad h_j(s_j, \gamma_j) := h_{\cup,j}(\gamma_j) + \left(\frac{s_j^2}{\sigma^2} + z_{2,j}\right)\gamma_j^{-1} + z_{1,j}\gamma_j - z_{3,j}\ln\gamma_j \quad (3.11)$$

we obtain

$$\frac{2}{\sigma^2}\phi_{\mathbf{z}}(\mathbf{u}) = \min_{\boldsymbol{\gamma}} \phi_{\mathbf{z}}(\boldsymbol{\gamma}, \mathbf{u}) \quad = \quad \frac{2}{\sigma^2}\left(\sum_{j=1}^q h_j^*(s_j) + \frac{1}{2}\|\mathbf{X}\mathbf{u} - \mathbf{y}\|^2 - \boldsymbol{\beta}^\top \mathbf{s}\right), \quad (3.12)$$

which is in standard form (section 3.5.1 fact 2) to be minimised using the iteratively reweighted least squares (IRLS) algorithm [Green, 1984] as introduced in chapter 2.3.1 and detailed for the inner loop minimisation of $\phi_{\mathbf{z}}(\mathbf{u})$ in section 3.5.5.

How the decomposition from above can be used to minimise $\phi(\boldsymbol{\gamma}, \mathbf{u})$ is summarised in algorithm 3.1. We will take a more detailed look at the outer and inner loop in the following.

3.5.4 Outer loop using the Lanczos algorithm

Outer loop updates of $\mathbf{z}_{1/2}$ require the computation of $\boldsymbol{\nu} = \sigma^2\mathrm{dg}(\mathbf{B}\mathbf{A}^{-1}\mathbf{B}^\top) = \mathbb{V}_Q[\mathbf{s}|\mathcal{D}]$, or equivalently all variances of the current Gaussian approximation to the model for fixed widths $\boldsymbol{\gamma}$. For large numbers of variables n, the variances $\boldsymbol{\nu}$ of the Gaussian model can be estimated by the Lanczos algorithm [Lanczos, 1950, Schneider and Willsky, 2001] as mentioned in chapter 2.5.4 and detailed in algorithm 3.2. In the absence of simple sparsity structure of $\mathbf{A}$, the Lanczos

***Algorithm 3.2** Lanczos tridiagonalisation algorithm*

```
Require: symmetric operator $\mathbf{A} \in \mathbb{R}^{n\times n}$, initial $\mathbf{q} \in \mathbb{R}^n$, $\mathbf{q}^\top\mathbf{q} = 1$ and empty $\mathbf{Q} = []$
  $\mathbf{v} \leftarrow \mathbf{Aq}$
  for $i = 1, 2, .., k$ do
    $\alpha_i \leftarrow \mathbf{q}^\top\mathbf{v}$
    $\mathbf{r} \leftarrow \mathbf{v} - \alpha_i\mathbf{q}$
    if $i > 1$ then
      $\mathbf{r} = \mathbf{r} - \mathbf{QQ}^\top\mathbf{r}$, reorthogonalise using Gram-Schmidt
    end if
    $\beta_i \leftarrow \sqrt{\mathbf{r}^\top\mathbf{r}}$, stop if too small
    if $i > 1$ then
      $e_i = \sqrt{\alpha_i - d_{i-1}^2}$, $d_i \leftarrow \frac{\beta_i}{e_i}$
    else
      $e_i = \sqrt{\alpha_i}$, $d_i \leftarrow \frac{\beta_i}{e_i}$
    end if
    $\mathbf{Q} \leftarrow [\mathbf{Q}, \mathbf{q}]$, include new Lanczos vector
    if $i < k$ then
      $\mathbf{v} \leftarrow \mathbf{q}$, $\mathbf{q} \leftarrow \frac{1}{\beta_i}\mathbf{r}$, $\mathbf{v} \leftarrow \mathbf{Aq} - \beta_i\mathbf{v}$
    end if
  end for
  $\mathbf{T} \leftarrow \begin{bmatrix} \alpha_1 & \beta_1 & & 0 \\ \beta_1 & \alpha_2 & \ddots & \\ & \ddots & \ddots & \beta_{k-1} \\ 0 & & \beta_{k-1} & \alpha_k \end{bmatrix}$, $\mathbf{L} \leftarrow \begin{bmatrix} e_1 & 0 & & 0 \\ d_1 & e_2 & \ddots & \\ & \ddots & \ddots & 0 \\ 0 & & d_{k-1} & e_k \end{bmatrix}$
Ensure: $\mathbf{Q} \in \mathbb{R}^{n\times k}$, $\mathbf{Q}^\top\mathbf{Q} = \mathbf{I}$, $\mathbf{Q}^\top\mathbf{AQ} = \mathbf{T}$, $\mathbf{LL}^\top = \mathbf{T}$
```

The iterative Lanczos procedure after Cornelius Lanczos allows to compute eigenvalues and eigenvectors of square matrices $\mathbf{A}$*. As an extension to the power method, it builds an orthogonal basis of the Krylov subspace* $\{\mathbf{q}, \mathbf{Aq}, .., \mathbf{A}^{k-1}\mathbf{q}\}$ *using* k *matrix vector multiplications with* $\mathbf{A}$*. The procedure is fully scalable in* n *since* $\mathbf{A}$ *is only implicitly accessed through matrix vector multiplications. Storage requirements of the Lanczos algorithm are* $\mathcal{O}(n)$*; the Gram-Schmidt process needs* $\mathcal{O}(n \cdot k)$ *for the matrix* $\mathbf{Q}$*. Similarly, computation is dominated by the* k *matrix vector multiplications and* $\mathcal{O}(n \cdot k^2)$ *for the orthogonalisation.*

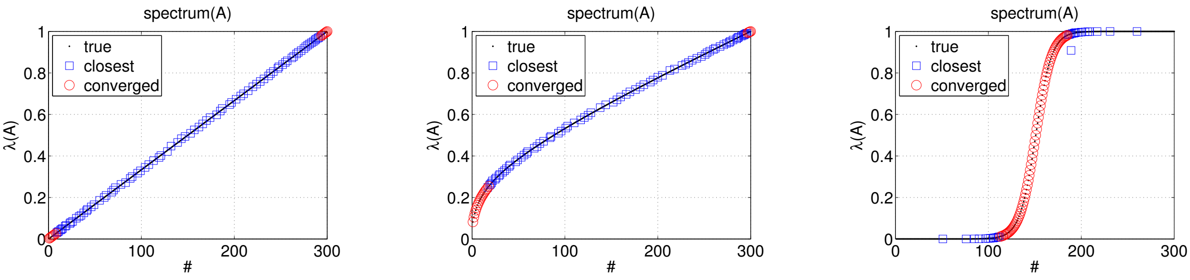


Figure 3.5: Convergence of Lanczos eigenvalues

Convergence of the eigenvalue/eigenvector pairs for symmetric matrices $\mathbf{A} \in \mathbb{R}^{n\times n}$*,* $n = 300$ *after* $k = 100$ *Lanczos iterations with different spectra. Left: linear spectrum. Centre: logarithmic spectrum. Right: sigmoid spectrum. The plot shows the exact eigenvalues along with converged Lanczos eigenvalue estimates (red) and not yet converged estimates (blue). Convergence happens from the smallest and largest eigenvalue inwards (linear, logarithmic) and preferably at places with large spectral gap (logarithmic, sigmoid).*

procedure yields a generic variance estimate. From part 4 of the theorem in 3.4, we know that the variances $\boldsymbol{\nu}$ can be bounded by the variational parameters $\boldsymbol{\gamma}$ using $\sigma^2\boldsymbol{\gamma} \succeq \boldsymbol{\nu} \succeq \mathbf{0}$.

In a nutshell, the precision matrix $\mathbf{A}$ is iteratively approximated by a low-rank representation $\mathbf{QTQ}^\top$, $\mathbf{Q} \in \mathbb{R}^{n\times k}$ orthonormal, $\mathbf{T} \in \mathbb{R}^{k\times k}$ tridiagonal, and $k \ll n$, where the eigenvalue/eigenvector pairs $(\theta_i, \mathbf{w}_i)$ of $\mathbf{T} = \mathbf{W\Theta W}^\top$ rapidly converge to eigenvalue/eigenvector pairs $(\omega_i, \mathbf{v}_i)$ of $\mathbf{A} = \mathbf{V\Omega V}^\top$. More specifically, convergence happens simultaneously from the smallest and largest eigenvalue inwards roughly ordered by the spectral gap between consecutive eigenvalues [Golub and van Loan, 1996, § 9.1.4] as illustrated by figure 3.5. Every iteration (out of the k iterations) requires only a single matrix vector multiplication with $\mathbf{A}$.

By $\mathbf{A}^{-1} \approx \mathbf{QT}^{-1}\mathbf{Q}^\top$, we can iteratively estimate $\boldsymbol{\nu} \approx \sigma^2 \mathrm{dg}(\mathbf{BQT}^{-1}\mathbf{Q}^\top\mathbf{B}^\top) =: \hat{\boldsymbol{\nu}}$ using the Lanczos procedure (algorithm 3.2). Starting from $\mathbf{w} = \hat{\boldsymbol{\nu}} = \mathbf{0}$, and inserting the recurrence

$$\mathbf{w} \leftarrow \frac{\mathbf{Bq} - d_{k-1}\mathbf{w}}{e_k},\ \hat{\boldsymbol{\nu}} \leftarrow \hat{\boldsymbol{\nu}} + \sigma^2 \mathbf{w} \odot \mathbf{w}$$

right after the inclusion of the new Lanczos vector yields the componentwise monotonically increasing estimator $\hat{\boldsymbol{\nu}}$ of the Gaussian variance $\boldsymbol{\nu}$. In this usage, the Lanczos algorithm can be thought of as solving many linear system in parallel, with the same $\mathbf{A}$ but different right hand sides.

Lanczos implementations for large n are not straightforward due to loss of orthogonality in the matrix $\mathbf{Q}$. As a consequence, practical Lanczos codes require an explicit Gram-Schmidt orthogonalisation [Golub and van Loan, 1996, § 9.2]. Ironically, it is the rapid convergence of the eigenvalues of $\mathbf{T}$ to the eigenvalues of $\mathbf{A}$ that causes the numerical problems [Paige, 1976, Parlett and Scott, 1979]. Re-orthogonalisation is not only computationally intense $\mathcal{O}(n\,k^2)$ but also requires significant memory $\mathcal{O}(n\,k)$. Thus, the algorithm can be run with moderate k only, significantly underestimating many components in $\hat{\boldsymbol{\nu}}$. This inaccuracy seems to be unavoidable: we are not aware of a general bulk variances estimator improving on Lanczos, and variances are required to drive any algorithm for $\min_{\boldsymbol{\gamma}} \phi$.

Importantly, systematic underestimation of $\boldsymbol{\nu}$ by $\hat{\boldsymbol{\nu}}$ does not seem to harm our algorithm in practise if used in the experimental design loop [Seeger, 2010a]. It appears that the design scores for the most promising candidates are accurately estimated relative to each other, even though only a small number of Lanczos vectors k is used to approximate $\mathbf{A}$. Inaccurate variances mean that $\min_{\mathbf{u}} \phi_{\mathbf{z}}(\boldsymbol{\gamma}, \mathbf{u})$ is not exactly tangent to $\phi(\boldsymbol{\gamma})$ at the current $\boldsymbol{\gamma}$ after an outer loop update. However, the (inner loop) minimisation is accurate, since mean computations by conjugate gradients are required only. Given the apparent intractability of the variance computation, this is a critical feature of our decoupling approach. Compared to other tractable inference approximations, where many dependencies are ruled out up front independent of the data, e.g. by factorisation assumptions in structured mean field, our approximation is fully data-dependent, with the extremal covariance eigenvectors being homed in by Lanczos similar PCA.

As a further consequence of the Lanczos approximation, our analytical convergence and convexity results are challenged: convexity can be compromised by the approximate calculation of $\boldsymbol{\nu}$, however convergence of the double loop algorithm can analytically be established if a fixed number of converged smallest eigenvector/eigenvalue pairs are used [Seeger, 2010a] instead of all k Lanczos vectors in $\mathbf{Q}$.

3.5.5 Inner loop by IRLS using conjugate gradients

The inner loop criterion as stated in equation 3.12

$$\phi_{\mathbf{z}}(\mathbf{u}) = \sum_{j=1}^{q} h_j^*(s_j) + \frac{1}{2}\|\mathbf{Xu} - \mathbf{y}\|^2 - \boldsymbol{\beta}^\top \mathbf{s} \tag{3.13}$$

is a sum of a quadratic and a decoupled part. Let us consider the implicitly defined 1-dimensional functions $h_j^*(s_j)$ (see equation 3.11) as simple for now and let us assume, we have the first two derivatives $\frac{\mathrm{d}}{\mathrm{d}s}h_j^*(s_j)$ and $\frac{\mathrm{d}^2}{\mathrm{d}s^2}h_j^*(s_j)$ available. In fact, the inner loop optimisation has the same structure as a MAP estimation or penalised least squares estimation problem of chapter 2.3.1 with $h_j^*(s_j) - \beta_j s_j$ taking the role of the penaliser. Thus, we can apply a variant of the Newton-Raphson algorithm to minimise $\phi_\mathbf{z}(\mathbf{u})$ called iteratively reweighted least squares (IRLS), see chapter 2.3.1. IRLS typically converges after a few Newton steps requiring the gradient and the Hessian in each

$$\begin{aligned}\mathbf{g} = \frac{\partial \phi_\mathbf{z}(\mathbf{u})}{\partial \mathbf{u}} &= \mathbf{B}^\top(\mathbf{h}' - \boldsymbol{\beta}) + \mathbf{X}^\top\mathbf{p},\ \mathbf{p} = \mathbf{X}\mathbf{u} - \mathbf{y},\ h_j' = \frac{\mathrm{d}}{\mathrm{d}s_j}h_j^*(s_j)\\ \mathbf{H} = \frac{\partial^2 \phi_\mathbf{z}(\mathbf{u})}{\partial \mathbf{u}\partial \mathbf{u}^\top} &= \mathbf{B}^\top\mathrm{dg}(\mathbf{h}'')\mathbf{B} + \mathbf{X}^\top\mathbf{X},\ h_j'' = \frac{\mathrm{d}^2}{\mathrm{d}s_j^2}h_j^*(s_j)\end{aligned}$$

to compute the Newton descent direction

$$\mathbf{d} = -\mathbf{H}^{-1}\mathbf{g} \Leftrightarrow \left(\mathbf{B}^\top\mathrm{dg}(\mathbf{h}'')\mathbf{B} + \mathbf{X}^\top\mathbf{X}\right)\mathbf{d} = \mathbf{B}^\top(\boldsymbol{\beta} - \mathbf{h}') - \mathbf{X}^\top\mathbf{r}$$

by solving an $n \times n$ linear system. Given useful structure in $\mathbf{X}$, $\mathbf{B}$ (such as sparsity or fast multiplication otherwise), this optimisation is scalable to very large sizes; the system is solved by (preconditioned) linear conjugate gradients (LCG). Next we compute a step size by conducting a 1-dimensional line search along $\mathbf{d}$. Evaluation of $\phi_\mathbf{z}$ along the line $\mathbf{u} + \lambda\mathbf{d}$ can be done in negligible time if $\mathbf{Bd}$, $\|\mathbf{Xd}\|^2$ and $\boldsymbol{\beta}^\top\mathbf{Bd} - \mathbf{p}^\top\mathbf{Xd}$ are precomputed

$$\phi_\mathbf{z}(\lambda) \overset{c}{=} \phi_\mathbf{z}(\mathbf{u} + \lambda\mathbf{d}) \overset{c}{=} \sum_{j=1}^{q} h_j^*(s_j + \lambda\mathbf{b}_j^\top\mathbf{d}) + \lambda^2\frac{\|\mathbf{Xd}\|^2}{2} - \lambda(\boldsymbol{\beta}^\top\mathbf{Bd} - \mathbf{r}^\top\mathbf{Xd})$$

so that no matrix vector multiplication (MVM) needs to be computed during the line search. Upon inner loop convergence, the minimiser $\mathbf{u}_* = \arg\min_\mathbf{u} \phi_\mathbf{z}(\mathbf{u})$ is the mean of the current posterior approximation $\mathbb{Q}(\mathbf{u}|\mathcal{D}, \boldsymbol{\gamma})$. Note that we did not use any operations other than MVMs with $\mathbf{X}$ and $\mathbf{B}$ making the approach fully scalable if these can be done efficiently.

For Laplace potentials and the $\phi_\cup^{(2)}(\boldsymbol{\gamma})$ bound, the scalar operations have a simple analytic form: $h_j(\gamma_j) = \tau_j^2\gamma_j$ and $h_j^*(s_j) = \sigma\tau_j\sqrt{\sigma^2 z_{2,j} + s_j^2}$. However, for other potentials such as the cumulative logistic (see table 3.1), we are not aware of an analytic expression for $h_j(\gamma_j)$. Since h_j and h_j^* are defined by scalar convex minimisations, all terms can be computed implicitly whenever required using Newton minimisation in one dimension and lookup tables. A generic implementation based on $g_j(x_j) = \ln\mathcal{T}_j(s_j)$, $x_j = s_j^2$, $g_j'(x_j)$ and $g_j''(x_j)$ alone, is provided in appendix E.4. Even with many implicitly defined h_j^*, the inner loop can be minimised efficiently because the $h_j^*(s_j)$ computations can be vectorised or parallelised straightforwardly.

Log-concave potentials

For all log-concave potentials such as logistic and cumulative logistic, the inner loop computations can be simplified considerably because of the simple relation

$$\begin{aligned}h_j^*(s_j) &= \beta_j\varsigma_j - \sigma^2 g(\varsigma_j), & g_j(s_j) &= \ln\mathcal{T}_j(s_j), & \varsigma_j &= \mathrm{sign}(s_j)\sqrt{s_j^2 + \sigma^2 z_{2,j}}\\ h^{*\prime}(s) &= \left[\beta - \sigma^2 g'(\varsigma)\right]\frac{s}{\varsigma}, & h^{*\prime\prime}(s) &= \left[\beta - \sigma^2\left(g'(\varsigma) + \frac{s^2\varsigma}{\nu}g''(\varsigma)\right)\right]\frac{\nu}{\varsigma^3}\end{aligned}$$

that we derive in appendix E.5. As a consequence, for the evaluation of $h_j^*(s_j)$ we only need to know the log potential $\ln\mathcal{T}_j(s_j)$; there is no need to deal with $h_j(\gamma_j)$ at any time in the algorithm. The minimum value γ_j needed for the outer loop update admits a similar expression (computed in appendix E.5)

$$\gamma_j = \frac{\varsigma_j}{\beta_j - \sigma^2 g_j'(\varsigma_j)} = \frac{s_j}{h_j^{*\prime}(s_j)},\ g_j(s_j) = \ln\mathcal{T}_j(s_j),\ \varsigma_j = \sqrt{s_j^2 + \sigma^2 z_{2,j}}.$$

Again, there is no need to deal with $h_j(\gamma_j)$ – only $g_j(s_j) = \ln \mathcal{T}_j(s_j)$ and its derivatives $g_j'(s_j)$ and $g_j''(s_j)$ need to be known.

3.5.6 Properties of the algorithm

In the following, we look at the double loop algorithm from a more general perspective and describe the precise relationship to MAP estimation. Furthermore, we discuss some known statistical features related to sparse estimation along with computational properties of the algorithm.

MAP estimation versus inference

The optimisation problems to compute MAP estimator $\hat{\mathbf{u}}_{\text{MAP}}$ (see chapter 2.5.6) and the posterior mean estimator $\hat{\mathbf{u}}_{\text{VB}}$ in the inner loop (IL) of our variational relaxation (see $\phi_{\mathbf{z}}(\mathbf{u})$ in section 3.5.5) have the same IRLS structure if we employ the $\phi_{\cup}^{(2)}(\gamma)$ bound for log-concave potentials, where $\mathbf{z}_1 = \mathbf{z}_3 = \mathbf{0}$ and use for $h_j^*(s_j)$ the expression from appendix E.5:

$$
\begin{aligned}
\hat{\mathbf{u}} &= \arg\min_{\mathbf{u}} \frac{1}{2} \|\mathbf{X}\mathbf{u} - \mathbf{y}\|^2 + \rho(\mathbf{s}),\ \mathbf{s} = \mathbf{B}\mathbf{u} \\
\rho_{\text{MAP}}(\mathbf{s}) &= -\sigma^2 \sum_{j=1}^{q} \ln \mathcal{T}_j(s_j) = -\sigma^2 \ln \mathcal{T}(\mathbf{s}) \\
\rho_{\text{IL}}(\mathbf{s}) &= h^*(\mathbf{s}) - \boldsymbol{\beta}^\top \mathbf{s} = \boldsymbol{\beta}^\top(\boldsymbol{\varsigma} - \mathbf{s}) + \rho_{\text{MAP}}(\boldsymbol{\varsigma}),\ \boldsymbol{\varsigma} = \text{sign}(\mathbf{s}) \odot \sqrt{\mathbf{s}^2 + \boldsymbol{\nu}},\ \boldsymbol{\nu} = \sigma^2 \mathbf{z}_2.
\end{aligned}
$$

First, for $\boldsymbol{\nu} = \mathbf{0}$, we exactly recover MAP estimation. Second, the larger the marginal variances ν_j, the less $h_j^*(s_j)$ depends on s_j. In other words, the marginal variances ν_j smoothly interpolate between MAP estimation and least squares estimation $\hat{\mathbf{u}}_{\text{LS}} = \arg\min_{\mathbf{u}} \frac{1}{2} \|\mathbf{X}\mathbf{u} - \mathbf{y}\|^2$. The relative trade-off between the two is adaptively computed in the outer loop; depending on the data.

Now, we can also understand the effect of underestimating marginal variances in the Lanczos algorithm in the outer loop (section 3.5.4): the variational Bayesian inference relaxation mean estimate is biased towards the posterior mode.

As a consequence, every inner loop iteration solves a "smoothed" MAP estimation problem and every outer loop adaptively updates the penaliser $h^*(\mathbf{s})$ by recomputing $\boldsymbol{\nu}$. Therefore, variational inference can be summarised as executing several MAP iterations with adaptive data-driven shrinkage of coefficients s_j. The term selective shrinkage was first employed by Ishwaran and Rao [2005] in bioinformatics.

Sparse linear models and experimental design

Let us look at the special case of the sparse linear model (SLM) with $\mathbf{B} = \mathbf{I}$ and Laplace potentials $-\ln \mathcal{T}(\mathbf{s}) = \frac{\tau}{\sigma} \|\mathbf{s}\|_1$, $\boldsymbol{\beta} = \mathbf{0}$ to gain some understanding of our variational inference relaxation in the context of sparse estimation. The respective ρ-penalised least squares problems for MAP estimation and the inner loop in variational inference are

$$
\hat{\mathbf{u}} = \arg\min_{\mathbf{u}} \frac{1}{2\sigma\tau} \|\mathbf{X}\mathbf{u} - \mathbf{y}\|^2 + \rho\left(\mathbf{u}\right),\ \rho_{\text{MAP}}(\mathbf{u}) = \|\mathbf{u}\|_1,\ \rho_{\text{VB}}(\mathbf{u}) = \min_{\mathbf{z}} \left\| \sqrt{\mathbf{u}^2 + \sigma^2 \mathbf{z}} \right\|_1 - \phi_{\cap}^*(\mathbf{z}),
$$

where the variational penaliser $\rho_{\text{VB}}(\mathbf{u})$ is only implicitly defined using $\phi_{\cap}^*$ the Legendre-Fenchel dual of $\boldsymbol{\gamma}^{-1} \mapsto \ln|\mathbf{A}|$: $\phi_{\cap}^*(\mathbf{z}) = \min_{\boldsymbol{\gamma}^{-1}} \mathbf{z}^\top \boldsymbol{\gamma}^{-1} - \ln|\mathbf{A}|$. Both approaches are instances of shrinkage estimators, i.e. $\mathbf{u}$ is shrunk towards zero as opposed to ordinary least squares estimation (see chapter 2.2.1). The L_1-norm in MAP estimation yields sparse solutions with many components being zero, since the minimum $\hat{\mathbf{u}}$ lies at a corner of the L_1-ball. On top of that, our variational inference relaxation applies shrinkage in an adaptive way depending on the marginal variances $\boldsymbol{\nu} = \mathbb{V}_{\mathrm{Q}}[\mathbf{u}|\mathcal{D}]$: for model parameters with small variance, the shrinkage

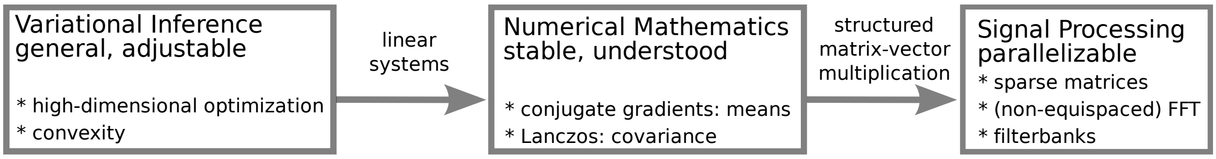


Figure 3.6: Reductions in variational inference

effect is larger, high variance leads to smaller penalty. Underestimation of $\boldsymbol{\nu}$ due to the Lanczos procedure leads to more sparsity in the estimate $\hat{\mathbf{u}}$.

Exact sparsity is certainly a computationally valuable property allowing to scale inference up to large models, however whether it is statistically appropriate depends on the application. In Bayesian approaches [Tipping, 2001, Wipf and Nagarajan, 2008], sparsity is equivalent to $\gamma_j = 0$ for some variational parameters and hence vanishing marginal variance $\nu_j = 0$ since $\nu_j \leq \sigma^2\gamma_j$, from theorem 4 of section 3.4. Zero variance or equivalently absolute certainty is very dangerous because not only γ_j is clamped but also all correlations between s_j and other components s_i become zero. The posterior distribution $\mathbb{Q}(\mathbf{u}|\mathcal{D})$ only exists in the orthogonal complement of the space spanned by the columns of $\mathbf{B}_J$ with $\gamma_j = 0$. Especially, in the experimental design context, where a trade-off between exploration and exploitation has to be found, it is problematic to rule out potentials early, because they cannot be explored later.

Scalability and complexity

The scalability of our algorithm comes from a number of appropriate *reductions* illustrated in figure 3.6. On the first level, the complicated inference problem (high-dimensional non-Gaussian integration) is relaxed to a convex program (variational approach). The corresponding optimisation problem is decoupled in the double loop algorithm: inner loop iterations reduce to the estimation of means $\mathbb{E}_{\mathbb{Q}}[\mathbf{u}|\mathcal{D}]$ in a linear-Gaussian model with LCG, and IRLS. The outer loop computes Gaussian variances $\mathbb{V}_{\mathbb{Q}}[\mathbf{s}|\mathcal{D}]$ by the Lanczos algorithm. On a higher level, we fit a sequence of Gaussian models to the exact non-Gaussian posterior. Hence, both inner and outer loops consist of standard algorithms from numerical linear algebra, routinely employed for very large systems. These naturally reduce to matrix-vector multiplications (MVMs). As a result, the inference algorithm is as fast as the MVMs with $\mathbf{X}$ and $\mathbf{B}$ rendering computations as scalable as MAP estimation. Therefore, exploitable structure in the system matrices $\mathbf{X}$ and $\mathbf{B}$ in terms of fast MVMs is crucial for our algorithm to be scalable to large numbers of variables n. The cost of an MVM with a sparse matrix is linear in the number of non-zeros, an MVM with a Fourier matrix demands $\mathcal{O}(n \cdot \ln n)$ and a wavelet transform requires $\mathcal{O}(n)$. Otherwise, our application to trajectory design for magnetic resonance imaging, where $n = 256^2$, $q \approx 3n$, $m = \frac{1}{4}n$ in chapter 6 would be impossible to deal with. Consequently, the computational complexity of the algorithm is measured in number of MVMs needed, and can be related to MAP estimation and a naive approach to minimising $\phi(\boldsymbol{\gamma})$.

Recall that n is the number of latent variables, m the number of Gaussian, and q the number of non-Gaussian potentials. Further, we denote by k the number of Lanczos iterations in outer loop updates, by N_{CG} the number of LCG iterations to solve a system with $\mathbf{A}$, and by N_{Newt} the number of Newton steps for IRLS. The computational complexities of the double loop algorithm, MAP estimation and alternative minimisation schemes is contrasted in table 3.2.

While the means of a large linear-Gaussian model can be estimated by a single linear system, the variances are much harder to obtain. In fact, we do not know of a general bulk variance estimator which is as accurate as LCG, but not vastly more expensive. To understand the rationale behind our algorithm, note that the computation of $\nabla_{\boldsymbol{\gamma}}\phi$ is as difficult as the estimation of $\mathbf{z}$. Our algorithm requires these expensive steps only a few times (usually 4 or 5 outer loop iterations are sufficient), since they are kept out of the inner loop, where most of the progress is made. In contrast, most standard gradient-based optimisers require many evaluations of

algorithm		# MVMs	storage
full Newton for MAP		$N_{\text{Newt}} \cdot N_{\text{CG}}$	$\mathcal{O}(m+n+q)$
one coordinate descent step in ϕ		$q \cdot N_{\text{CG}}$	$\mathcal{O}(m+n+q)$
one exact $\nabla_{\gamma}\phi$		$q \cdot N_{\text{CG}}$	$\mathcal{O}(m+n+q)$
one approx $\nabla_{\gamma}\phi$		$k + N_{\text{CG}}$	$\mathcal{O}(k \cdot n+q)$
double	inner: $\hat{\mathbf{u}} = \arg\min_{\mathbf{u}} \phi_{\mathbf{z}}(\mathbf{u}) = \mathbb{E}_{\mathrm{Q}}[\mathbf{u}\|\mathcal{D}]$	$N_{\text{Newt}} \cdot N_{\text{CG}}$	$\mathcal{O}(n+q)$
loop	outer: $\mathbf{z} = \mathrm{dg}(\mathbf{B}\mathbf{A}^{-1}\mathbf{B}^{\top}) = \sigma^{-2}\mathbb{V}_{\mathrm{Q}}[\mathbf{s}\|\mathcal{D}]$	k	$\mathcal{O}(k \cdot n+q)$

Table 3.2: Computational complexity of alternative algorithms

$\nabla_{\gamma}\phi$ to converge. As discussed below, our decomposition also means that the variances can be estimated rather poorly, while still obtaining a practically useful algorithm.

Empirically, $N_{\text{Newt}} \approx 10$ for our inner loops, and we never run more than 5 outer loop iterations, typically 1 or 2 only. Lanczos codes come with additional costs to keep $\mathbf{Q}$ orthonormal, up to $\mathcal{O}(n \cdot k^2)$. The table shows that a naive minimisation of $\phi(\gamma)$ is not scalable, since we have to solve $\mathcal{O}(q)$ $n \times n$ linear systems for a single gradient step. While MAP estimation is faster in practise, its scaling differs from our algorithm's only by a moderate constant factor.

3.6 Implementation

In our implementation, we use the bounding technique with objective $\phi_{\mathrm{U}}^{(2)}(\gamma)$ (equation 3.9). We offer an entire toolbox for generalised linear model inference and estimation (`glm-ie`) whose code can be obtained from `http://mloss.org/software/view/269/`. The code is fully compatible to both Matlab 7.x[5] and GNU Octave 3.2.x[6]. It has been thouroughly tested and verified. Its modular and generic structure entail extensibility and quite a big range of applications.

3.6.1 The `glm-ie` toolbox

The `glm-ie` toolbox handles generalised linear models of the general form detailed in chapter 2.5. Both MAP or PLS estimation (chapter 2.7) and variational Bayesian inference are covered.

The toolbox contains the following objects:

- Potential functions $\mathcal{T}(s)$: They have to be positive, symmetrisable and super-Gaussian. An implementation requires $\ln \mathcal{T}(s)$, its first two derivatives $[\ln \mathcal{T}]'(s)$, $[\ln \mathcal{T}]''(s)$ and the symmetry parameter β. We offer Gaussian, Laplacian, Sech-square, Logistic, Exponential power and Student's t potentials.

- Penalty functions $\rho(s)$: The have to be continuously differentiable; convexity is not required but makes the PLS problem much simpler. An implementation requires the evaluation of $\rho(s)$ and its first two derivatives $\rho'(s)$, $\rho''(s)$. We offer a penalty function derived from a potential function that allows to express the inner loop as a PLS problem. Other penalties comprise the logarithmic, quadratic, power and zero penalisers.

- PLS solvers: MAP, PLS and the inner loop require optimisation routines. We use a generic interface implementing a CG solver, a CG solver with backtracking line search, a quasi-Newton algorithm and a truncated Newton procedure.

- Matrix operators: The algorithm uses MVMs as building blocks. Therefore, we have many matrix objects implemented such as finite difference, convolution, wavelet and Fourier transform matrices.

More details and illustrating examples can be found in the documentation of the toolbox.

[5] The MathWorks, `http://www.mathworks.com/`

[6] The Free Software Foundation, `http://www.gnu.org/software/octave/`

3.7 Bayesian active learning for binary classification

In the following, we apply the scalable algorithm to a large-scale binary classification task on datasets frequently used in machine learning research.

Probabilistic classification is a special case of our generalised linear model framework. We use linear classifiers with cumulative logistic likelihoods (see figure 2.2b and chapter 4)

$$\mathbb{P}(c_j|\mathbf{u},\mathbf{b}_j) = \frac{1}{1+\exp(-c_j \cdot \frac{\tau_{\text{sig}}}{\sigma}\mathbf{b}_j^\top\mathbf{u})} = \mathcal{T}_j(s_j;c_j),\ \mathbf{s}=\mathbf{Bu},$$

where $\mathbf{u}\in\mathbb{R}^n$ denotes the classifier weights, $\mathbf{b}_j\in\mathbb{R}^n$ contains the feature vector for data point j, $c_j\in\{\pm1\}$ is the class label and $\tau_{\text{sig}}>0$ is a scaling parameter. The matrix $\mathbf{B}=[\mathbf{b}_1,..,\mathbf{b}_q]^\top\in\mathbb{R}^{q\times n}$ contains the q feature vectors $\mathbf{b}_j$ as rows and the vector $\mathbf{c}\in\mathbb{R}^q$ collects respective labels c_j of the training set of size q. For the remainder, we concentrate on a Gaussian weight prior $\mathbb{P}(\mathbf{u})=\mathcal{N}(\mathbf{u}|\mathbf{0},\sigma^2\mathbf{I})$ yielding $\mathbf{X}=\mathbf{I}$, $\mathbf{y}=\mathbf{0}$ and $m=n$. However, if the number of features n is much larger than the training set size, a sparsity prior might become appropriate leading to $\mathbf{X}=[]$, $\mathbf{y}=[]$, $\sigma=1$ and $m=0$; formally, we append $\mathbf{I}$ to $\mathbf{B}$ increasing q by n and add n Laplacian sparsity potentials $\mathcal{T}_j(s_j)=\exp(-\frac{\tau_{\text{lap}}}{\sigma}|s_j|)$. In our experiments, we use both sparsity and Gaussian weight priors but concentrate on the Gaussian case to simplify notation.

The goal of active learning is to reduce the amount of labels c_j needed for an accurate prediction by actively selecting the data points $\mathbf{b}_j$ from a candidate set $\mathcal{J}$ for which the labels c_j are to be acquired. We summarise all candidates $\mathbf{b}_j$, $j\in\mathcal{J}$ (also the ones already included in the model) in a big matrix $\mathbf{B}_\mathcal{J}$ so that $\mathbf{B}$ contains a subset of the rows of $\mathbf{B}_\mathcal{J}$. We adopt a sequential (greedy) approach, where in each block K new candidates are chosen from $\mathcal{J}$. The basis for active learning or Bayesian experimental design is the current representation of uncertainty in the classifier weights – the Bayesian posterior

$$\mathbb{Q}(\mathbf{u})=\mathcal{N}(\mathbf{u}|\mathbf{m},\mathbf{V})\approx\mathbb{P}(\mathbf{u}|\mathbf{c})\propto\mathbb{P}(\mathbf{u})\prod_{j=1}^{q}\mathbb{P}(c_j|\mathbf{u},\mathbf{b}_j)=\mathcal{N}(\mathbf{u}|\mathbf{0},\sigma^2\mathbf{I})\prod_{j=1}^{q}\mathcal{T}_j(s_j;c_j)$$

as approximated by the double loop algorithm of section 3.5. More specifically, the active learning decision about which candidate to include next is entirely based on the approximate posterior marginals

$$\mathbb{P}(s_j|\mathbf{c})\approx\mathbb{Q}(s_j)=\mathcal{N}(s_j|\mu_j,\sigma^2\rho_j),\ \mu_j=\mathbf{m}^\top\mathbf{b}_j,\ \rho_j=\frac{1}{\sigma^2}\mathbf{b}_j^\top\mathbf{V}\mathbf{b}_j.$$

The next subsection explains how to include a new potential $\mathcal{T}_j(s_j;c_j)$ into the model.

3.7.1 Non-Gaussian potential inclusion

If we wish to include the potential $\mathcal{T}_j(s_j;c_j)$ into posterior of the current model, we have to assign a new variational parameter γ_j for the respective potential. The lower bound to $\mathbb{P}(\mathcal{D}\cup\{\mathbf{b}_j,c_j\})$ seen as a function of γ_j is given by

$$\mathbb{P}(\mathcal{D}\cup\{\mathbf{b}_j,c_j\})\overset{c}{\geq}e^{-h_j(\gamma_j)/2}\mathbb{E}_{\mathbb{Q}(\mathbf{u})}\left[e^{\sigma^{-2}(\beta_j s_j-s_j^2/(2\gamma_j))}\right]\propto e^{-\phi_j(\gamma_j)/2}$$

up to a constant not depending on γ_j, where we treat all other variational parameters as fixed. After some algebra, we obtain

$$\phi_j(\gamma_j)=h_j(\gamma_j)+\log\kappa_j-\frac{(\mu_j+\rho_j\beta_j)^2}{\sigma^2\rho_j\kappa_j},\ \kappa_j:=1+\frac{\rho_j}{\gamma_j},\tag{3.14}$$

where $\mathbb{Q}(s_j)=\mathcal{N}(s_j|\mu_j,\sigma^2\rho_j)$. Therefore, the novel γ_j is computed as $\gamma_j^\star=\arg\min_{\gamma_j}\phi_j(\gamma_j)$ using standard $1d$ Newton techniques from convex minimisation.

The marginals $(\boldsymbol{\mu}, \boldsymbol{\rho})_{\mathcal{J}}$ for all candidates from $\mathcal{J}$ are updated as: $\boldsymbol{\rho}'_{\mathcal{J}} = \boldsymbol{\rho}_{\mathcal{J}} - \frac{1}{\rho_j+\gamma_j}\mathbf{w} \odot \mathbf{w}$, $\boldsymbol{\mu}'_{\mathcal{J}} = \boldsymbol{\mu}_{\mathcal{J}} + \frac{\beta_j-\mu_j/\gamma_j}{\kappa_j}\mathbf{w}$, where $\kappa_j = 1 + \rho_j/\gamma_j$ and $\mathbf{w} = \mathbf{B}_{\mathcal{J}}\mathbf{A}^{-1}\mathbf{b}_j$ (one linear system). We use the solution to recompute ρ_j, μ_j, solve again for γ_j, and plug these back into $\boldsymbol{\mu}_{\mathcal{J}}, \boldsymbol{\rho}_{\mathcal{J}}$. This corrects for Lanczos inaccuracies (especially since ρ_j is underestimated by the Lanczos procedure). Moreover, $\mathbf{u}'_* = \mathbf{u}_* \frac{\beta_j-\mu_j/\gamma_j}{\kappa_j}\mathbf{A}^{-1}\mathbf{b}_j$, and $\ln|\mathbf{A}'| = \ln|\mathbf{A}| + \ln\kappa_i$.

At the end of a block, we re-run our variational algorithm in order to update all variational parameters jointly (within a block, only γ_j for novel model potentials are updated). In practise, a single outer loop iteration suffices for these runs. Importantly, the first outer loop update comes for free, since the model marginals (part of $\boldsymbol{\mu}_{\mathcal{J}}$, $\boldsymbol{\rho}_{\mathcal{J}}$), $\mathbf{u}_*$, and $\log|\mathbf{A}|$ have been kept valid. Therefore, only a single Lanczos run per block is required. Finally, since variances are underestimated by Lanczos, it may happen that components in $\boldsymbol{\rho}_{\mathcal{J}}$ become negative within a block. Such components are simply removed, and if they correspond to model potentials, their marginals are recomputed by solving linear systems at the end of the block.

While there is some computational complexity to our scheme, note that the principal computational primitives are always the same: solving linear systems with $\mathbf{A}$, and variance estimation by Lanczos based on $\mathbf{A}$.

3.7.2 Active learning scores

Active learning can be done using a large variety of criteria. For an empirical review and collection of heuristics see Schein and Ungar [2007]. We use sequential Bayesian active learning, meaning that the scores for inclusion decisions are computed based on the marginals $\mathbb{Q}(s_j)$ of the posterior distribution. Given that, we can employ a host of different scores, and the particular ones used in our experiments (information gain IG and classifier uncertainty CU) could certainly be improved upon by heuristic experience with the task.

Our active learning algorithm starts with a posterior approximation based on randomly drawn instances. In the subsequent design phase, we sequentially include blocks of K data points each. If the task requires a large number of sequential inclusions, tractability is retained by choosing K large enough.

Each iteration consists of an initial Lanczos run to estimate marginal posterior moments, $K \geq 1$ inclusions (appending K new rows to $\mathbf{B}$), and a re-optimisation of all potential parameters $\boldsymbol{\gamma}$. Within a block, the marginals $\mathbb{Q}(s_j) = \mathcal{N}(s_j|\mu_j, \sigma^2\rho_j)$, $j \in J$ containing all model and candidate potentials, are kept valid at all times. Note that $\boldsymbol{\mu}_{\mathcal{J}} = \mathbf{B}_{\mathcal{J}}\mathbf{u}_*$ (since $\mathbf{u}_* = \mathbb{E}_{\mathbb{Q}(\mathbf{u})}[\mathbf{u}|\mathcal{D}]$), and that $\mathbf{B}$ is a part of $\mathbf{B}_J$. For larger K, our method runs faster, since the variational parameters $\boldsymbol{\gamma}$ are updated less frequently, while for smaller K, the more frequent refits to the non-Gaussian posterior may result in better sequential decisions.

Each inclusion within a block consists of scoring all remaining candidates, picking the winner, and updating the marginals $\boldsymbol{\mu}_{\mathcal{J}}, \boldsymbol{\rho}_{\mathcal{J}}$. Let $\mathbf{b}_j$ be a new candidate row of $\mathbf{B}$, and $s_j = \mathbf{b}_j^\top\mathbf{u}$. In our experiments, we use several design scores, based on the current (Gaussian) marginal $\mathbb{Q}(s_j)$: information gain IG and classifier uncertainty CU.

1. The classifier uncertainty score

$$CU(\mathbf{b}_j) \quad = \quad -\left|\mathbb{Q}(c_j = +1) - \frac{1}{2}\right|,$$

   prefers candidates with predictive probability $\mathbb{Q}(c_j = +1)$ close to $\frac{1}{2}$. We compute the required expectation

$$\mathbb{Q}(c_j) = \int \mathbb{Q}(s_j|\mathbf{c})\mathbb{P}(c_j|s_j)\mathrm{d}s_j = \int \mathcal{N}(s_j|\mu_j, \sigma^2\rho_j)\mathcal{T}_j(s_j; c_j = +1)\mathrm{d}s_j$$

   by Gaussian quadrature.

2. The information gain score (chapter 2.6.2, equation 2.26) is given by

$$IG(\mathbf{b}_j) \quad = \quad \sum_{c_j=\pm 1} \mathrm{Q}(c_j)\mathrm{KL}[\mathrm{Q}'(s_j;c_j) \,\|\, \mathrm{Q}(s_j)],$$

where $\mathrm{Q}'(s_j;c_j)$ is the new approximation to $\propto \mathrm{Q}(s_j)\mathcal{T}_j(s_j;c_j)$ after an additional potential $\mathcal{T}_j(s_j;c_j)$ has been included. If $\mathrm{Q}'(s_j) \propto \mathrm{Q}(s_j)e^{\sigma^{-2}(\beta_j s_j - \frac{1}{2}s_j^2/\gamma_j)}$ at the minimiser $\gamma_j^\star$, then

$$\mathrm{KL}[\mathrm{Q}' \,\|\, \mathrm{Q}] = \frac{1}{2}\left(\log\kappa_j + \frac{\rho_j}{\kappa_j}\left(\frac{(\beta_j - \mu_j/\gamma_j)^2}{\sigma^2\kappa_j} - \gamma_j^{-1}\right)\right),$$

which has to be computed for both label assumptions $c_j = \pm 1$, where $\beta_j = c_j \tau_{\mathrm{sig}}\sigma/2$.

Both scores are used in the following experiments.

3.7.3 Experiments

We use three standard datasets for binary classification[7], outlined in table 3.3. The feature vectors are sparse, and a MVM with the matrix $\mathbf{B}$ costs $\mathcal{O}(\#\mathrm{nz})$.

Dataset	q	q_+/q_-	n	# non-zeros
`a9a`	$32,561$	0.32	123	$451,592$
`real-sim`	$72,201$	0.44	$20,958$	$3,709,083$
`rcv1`	$677,399$	1.10	$42,736$	$49,556,258$

Table 3.3: Dimensionality of the considered datasets

We randomly select 16, 36 and 50 thousand instances for training; the rest is kept for testing. The hyperparameters τ_{sig}, σ^2, and τ_{lap} were determined on the full datasets, where τ_{lap} is only present if Laplacian potentials are used. Results are given in figure 3.7. We ran sparse logistic regression (with Laplace prior) on `a9a` only. As expected, our algorithm runs longer in this case, and is less tolerant w.r.t. larger block sizes K: the Laplace prior potential parameters have to be updated in response to new cases, in order to do their job properly. Although sparse classification improves on the Gaussian prior case beyond about 2800 cases, active learning works better with a Gaussian prior for fewer inclusions. This may be due to the case that the Lanczos variance estimation is exact for $q < k$, and in general more accurate in the Gaussian prior case. Over all sets, we see clear improvements of active learning with the classifier uncertainty score CU over random sampling of data cases. Somewhat surprisingly, the information gain score does much less well in the binary classification case.

3.8 Discussion

We have shown that a frequently used variational relaxation to Bayesian inference in super-Gaussian generalised linear models is convex if and only if the posterior is log-concave – variational inference is convex whenever MAP estimation is convex in the same model. The technique covers a wide class of models ranging from robust regression and classification to sparse linear modelling and complements the large body of work on efficient point estimation in sparse linear models. Our theoretical insights settle a long-standing question in approximate variational inference in continuous variable models and add details to the relationship between sparse estimation and sparse inference.

Further, we have developed a scalable double loop minimisation algorithm that runs orders of magnitude faster than previous coordinate descent methods, enhancing the scope for the Bayesian design methodology to large scales. This is achieved by decoupling the criterion

[7] `http://www.csie.ntu.edu.tw/~cjlin/libsvmtools/datasets/`

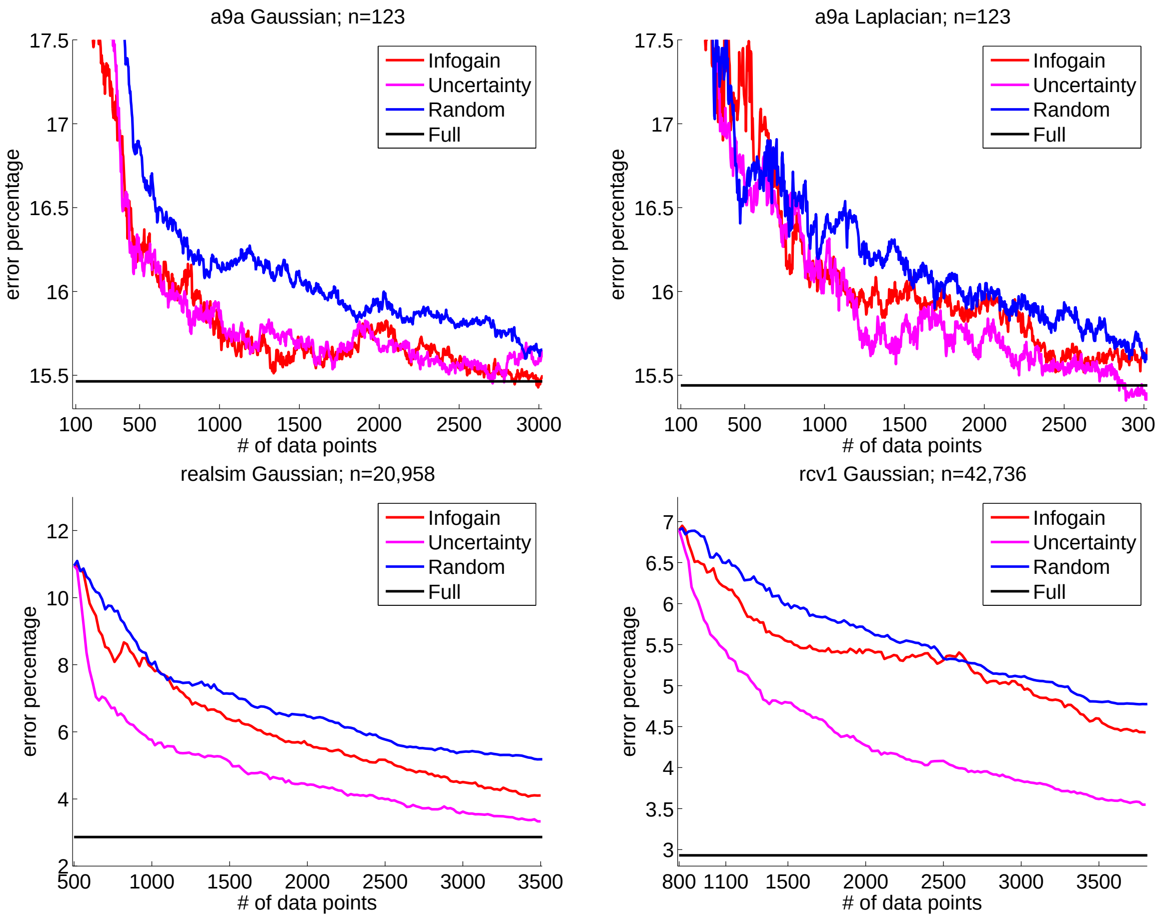


Figure 3.7: Classification errors for different design scores
Performance of information gain and classifier uncertainty versus random sampling (results on full training set also shown). We started the design phase after 100, 100, 500, 800 *randomly drawn initial cases respectively, all remaining training cases were candidates. The prior variance was set to* $\sigma^2 = 1$ *in all cases,* $\tau_{sig} = 1, 1, 3, 3$ *respectively.* $k = 80, 80, 750, 750$ *Lanczos vectors were computed for outer loop updates/candidate scoring. For* a9a*, we used design blocks of size* $K = 3$*, and* $K = 20$ *for the others.*

and using ideas from concave-convex programming. Computational efforts are reduced to fast algorithms known from estimation and numerical mathematics and exploiting fast MVMs with the structured matrices $\mathbf{X}$ and $\mathbf{B}$. Our generic implementation, can be run with any configuration of super-Gaussian, log-concave potentials using simple scalar minimisations, without any heuristics to be tuned.

From a graphical model perspective, our method reduces approximate inference in non-Gaussian (continuous variable) Markov random fields (MRFs) to repeated computations in Gaussian MRFs. In this context, we especially emphasise the importance of Gaussian marginal variance computations by the Lanczos algorithm. The considerable literature on Gaussian MRF techniques [Malioutov et al., 2006a,b] can be put to new use with our relaxation.

An interesting direction for future work is to find out what is so special about the chosen variational relaxation so that it leads to a scalable algorithm and to try and develop scalable variants of other approximate inference techniques.

Chapter 4

Gaussian Process Classification

We provide a comprehensive overview of many recent algorithms for approximate inference in Gaussian process models for probabilistic binary classification. The relationships between several approaches are elucidated theoretically, and the properties of the different algorithms are corroborated by experimental results. We examine both the quality of the predictive distributions and the suitability of the different marginal likelihood approximations for model selection (selecting hyperparameters) and compare to a gold standard based on MCMC. Interestingly, some methods produce good predictive distributions although their marginal likelihood approximations are poor. Strong conclusions are drawn about the methods: the expectation propagation algorithm is almost always the method of choice unless the computational budget is very tight. We also extend existing methods in various ways, and provide unifying code implementing all approaches.

Note that all derived inference algorithms are a special case of the generalised linear model framework of chapters 2.3, 2.4 by setting $\sigma = 1$, $\mathbf{B} = \mathbf{I}$, $\gamma = \sigma_n^2$ and formally substituting $\mathbf{X}^\top \mathbf{y} \leftarrow \mathbf{y}$ and $\mathbf{X}^\top \mathbf{X} \leftarrow \mathbf{K}^{-1}$ and that all analytical properties derived in chapter 3 carry over. The exposition is a revised and extended version of Nickisch and Rasmussen [2008] and details about the code are taken from Rasmussen and Nickisch [2010], `http://mloss.org/software/view/263/` and `http://gaussianprocess.org/gpml/code/`.

We start the chapter by introducing Gaussian processes in section 4.1 and show how they can be used in probabilistic classification models in section 4.2. Next, each of the sections 4.3, 4.4, 4.5, 4.6 and 4.8 describe a particular deterministic approximate inference method; the relation between them are reviewed in section 4.9. A sampling approach to approximate inference serving as gold standard is presented in section 4.10. Numerical implementation issues are discussed in section 4.11. We then empirically compare the approximate inference algorithms with each other and the gold standard in section 4.12 and draw an overall conclusion in section 4.13.

4.1 Introduction

Gaussian processes (GPs) can conveniently be used to specify prior distributions for Bayesian inference. In the case of regression with Gaussian noise, inference can be done simply in closed form, since the posterior is also a GP. For non-Gaussian likelihoods, such as, e.g. in binary classification, exact inference is analytically intractable.

One prolific line of attack is based on approximating the non-Gaussian posterior with a tractable Gaussian distribution. One might think that finding such an approximating GP is a well-defined problem with a largely unique solution. However, we find no less than three different types of solution in the recent literature: *Laplace approximation* (LA) [Williams and Barber, 1998], *expectation propagation* (EP) [Minka, 2001a] and *Kullback-Leibler divergence* (KL) minimisation [Opper and Archambeau, 2009] comprising *variational bounding* (VB) [Gibbs and MacKay, 2000, Jaakkola and Jordan, 1996] as a special case. Another approach is based on a factorial approximation, rather than a Gaussian [Csató et al., 2000].

Practical applications reflect the richness of approximate inference methods: LA has been used for sequence annotation [Altun et al., 2004] and prostate cancer prediction [Chu et al., 2005], EP for affect recognition [Kapoor and Picard, 2005], VB for weld cracking prognosis [Gibbs and MacKay, 2000], *Label regression* (LR) serves for object categorisation [Kapoor et al., 2007] and MCMC sampling is applied to rheumatism diagnosis by Schwaighofer et al. [2003]. Brain computer interfaces [Zhong et al., 2008] even rely on several (LA, EP, VB) methods.

We compare these different approximations and provide insights into the strengths and weaknesses of each method, extending the work of Kuss and Rasmussen [2005] in several directions: We cover many more approximation methods (VB, KL, FV, LR), put all of them in common framework and provide generic implementations dealing with both the logistic and the cumulative Gaussian likelihood functions and clarify the aspects of the problem causing difficulties for each method. We derive Newton's method for KL and VB. We show how to accelerate MCMC simulations. We highlight numerical problems, comment on computational complexity and supply runtime measurements based on experiments under a wide range of conditions, including different likelihood and different covariance functions. We provide deeper insights into the methods behaviour by systematically linking them to each other. Finally, we review the tight connections to methods from the literature on Statistical Physics, including the TAP approximation and TAPnaive.

The quantities of central importance are the quality of the probabilistic predictions and the suitability of the approximate marginal likelihood for selecting parameters of the covariance function (hyperparameters). The marginal likelihood for any Gaussian approximate posterior can be lower bounded using Jensen's inequality, but the specific approximation schemes also come with their own marginal likelihood approximations.

We are able to draw clear conclusions. Whereas every method has good performance under some circumstances, only a single method gives consistently good results. We are able to theoretically corroborate our experimental findings; together this provides solid evidence and guidelines for choosing an approximation method in practise.

4.2 Gaussian processes for binary classification

A GP prior over latent the function $f(\mathbf{x}) \sim \mathcal{GP}(m(\mathbf{x}), k(\mathbf{x}, \mathbf{x}'))$ in conjunction with a likelihood $\mathbb{P}(y_i|f_i)$, leads to a posterior process f_* that is conditioned on the data $(\mathbf{x}_i, y_i)_{i=1..n}$. In case $\mathbb{P}(y_i|f_i) = \mathcal{N}(f_i|y_i, \sigma^2)$ is Gaussian, the posterior process will again be a GP. As with generalised linear models, we can absorb every link function into the likelihood and can therefore model non-negativity along the lines of the *warped Gaussian process* of Snelson et al. [2004]. In geospatial statistics, this technique is known under the name *kriging* for *generalised linear spatial models* [Diggle et al., 1998].

Although most of the technical machinery is fully generic in the likelihood $\mathbb{P}(y_i|f_i)$, we concentrate on probabilistic binary classification based on Gaussian processes. Keep in mind that any of the likelihoods in figure 2.2 can be used. For a graphical model representation see figure 4.1 and for a 1d pictorial description consult figure 4.2. Given data points $\mathbf{x}_i$ from a domain $\mathcal{X}$ with corresponding class labels $y_i \in \{-1, +1\}$, one would like to predict the class membership probability $\mathbb{P}(y_*|\mathbf{x}_*, \mathbf{y}, \mathbf{X})$ for a test point $\mathbf{x}_*$. This is achieved by using a *latent function* f whose value is mapped into the unit interval by means of a sigmoid function $\mathrm{sig} : \mathbb{R} \to [0, 1]$ so that the class membership probability $\mathbb{P}(y = +1|\mathbf{x})$ can be written as $\mathrm{sig}(f(\mathbf{x}))$. The class membership probability must normalise $\sum_y \mathbb{P}(y|\mathbf{x}) = 1$, which leads to $\mathbb{P}(y = +1|\mathbf{x}) = 1 - \mathbb{P}(y = -1|\mathbf{x})$ and consequently to $\mathbb{P}(y|\mathbf{x}) = \mathrm{sig}(f(\mathbf{x}))^{\frac{1+y}{2}} - 1 + (1 - \mathrm{sig}(f(\mathbf{x})))^{\frac{1-y}{2}}$ (defining $0^0 = 1$). If the sigmoid function satisfies the point symmetry condition $\mathrm{sig}(t) = 1 - \mathrm{sig}(-t)$, the *likelihood* can be compactly written as

$$\mathbb{P}(y|\mathbf{x}) \quad = \quad \mathrm{sig}(y \cdot f(\mathbf{x})).$$

We consider two point symmetric sigmoids (see likelihood figure 2.2a)

$$\mathrm{sig}_{\mathrm{logit}}(t) := \frac{1}{1+e^{-t}} \quad \text{(cumulative logistic), and} \tag{4.1}$$

$$\mathrm{sig}_{\mathrm{probit}}(t) := \int_{-\infty}^{t} \mathcal{N}(\tau|0,1)\mathrm{d}\tau \quad \text{(cumulative Gaussian).} \tag{4.2}$$

The two functions are very similar at the origin (showing locally linear behaviour around $\mathrm{sig}(0) = 1/2$ with slope $1/4$ for $\mathrm{sig}_{\mathrm{logit}}$ and $1/\sqrt{2\pi}$ for $\mathrm{sig}_{\mathrm{probit}}$) but differ in how fast they will approach $0/1$ if t goes to infinity. Namely in the logarithmic domain, we have for large negative values of t the following asymptotics:

$$\mathrm{sig}_{\mathrm{logit}}(t) \approx \exp(-t) \quad \text{and} \quad \mathrm{sig}_{\mathrm{probit}}(t) \approx \exp(-\frac{1}{2}t^2 + 0.158t - 1.78), \text{ for } t \ll 0.$$

Linear decay of $\ln(\mathrm{sig}_{\mathrm{logit}})$ corresponds to a weaker penalty for wrongly classified examples than the quadratic decay of $\ln(\mathrm{sig}_{\mathrm{probit}})$.

For notational convenience, the following shorthands are used: the matrix $\mathbf{X} = [\mathbf{x}_1, \ldots, \mathbf{x}_n]$ of size $n \times d$ collects the training points, the vector $\mathbf{y} = [y_1, \ldots, y_n]^\top$ of size $n \times 1$ collects the target values and latent function values are summarised by $\mathbf{f} = [f_1, \ldots, f_n]^\top$ with $f_i = f(\mathbf{x}_i)$. Observed data is written as $\mathcal{D} = \{(\mathbf{x}_i, y_i)\, | i = 1, \ldots, n\} = (\mathbf{X}, \mathbf{y})$. Quantities carrying an asterisk refer to test points, i.e. $\mathbf{f}_*$ contains the latent function values for test points $[\mathbf{x}_{*,1}, \ldots, \mathbf{x}_{*,m}] = \mathbf{X}_* \subset \mathcal{X}$. Covariances between latent values $\mathbf{f}$ and $\mathbf{f}_*$ at data points $\mathbf{x}$ and $\mathbf{x}_*$ follow the same notation, namely $[\mathbf{K}_{**}]_{ij} = k(\mathbf{x}_{*,i}, \mathbf{x}_{*,j})$, $[\mathbf{K}_*]_{ij} = k(\mathbf{x}_i, \mathbf{x}_{*,j})$, $[\mathbf{k}_*]_i = k(\mathbf{x}_i, \mathbf{x}_*)$ and $k_{**} = k(x_*, x_*)$, where $[\mathbf{A}]_{ij}$ denotes the entry A_{ij} of the matrix $\mathbf{A}$.

Given the latent function f, the class labels are assumed to be Bernoulli distributed and independent random variables, which gives rise to a *factorial likelihood*, factorising over data points (see figure 4.1):

$$\mathbb{P}(\mathbf{y}|f) = \mathbb{P}(\mathbf{y}|\mathbf{f}) = \prod_{i=1}^{n} \mathbb{P}(y_i|f_i) = \prod_{i=1}^{n} \mathrm{sig}(y_i f_i) \tag{4.3}$$

A GP [Rasmussen and Williams, 2006] is a stochastic process fully specified by a *mean function* $m(\mathbf{x}) = \mathbb{E}[f(\mathbf{x})]$ and a positive definite *covariance function* $k(\mathbf{x}, \mathbf{x}') = \mathbb{V}[f(\mathbf{x}), f(\mathbf{x}')]$. This means that a random variable $f(\mathbf{x})$ is associated to every $\mathbf{x} \in \mathcal{X}$, so that for any set of inputs $\mathbf{X} \subset \mathcal{X}$, the joint distribution $\mathbb{P}(\mathbf{f}|\mathbf{X}, \boldsymbol{\theta}) = \mathcal{N}(\mathbf{f}|\mathbf{m}_0, \mathbf{K})$ is Gaussian with mean vector $\mathbf{m}_0$ and covariance matrix $\mathbf{K}$. The mean function and covariance functions may depend on additional *hyperparameters* $\boldsymbol{\theta}$. For notational convenience we will assume $m(x) \equiv 0$ throughout. Thus, the elements of $\mathbf{K}$ are $K_{ij} = k(\mathbf{x}_i, \mathbf{x}_j, \boldsymbol{\theta})$.

By application of Bayes' rule, one gets an expression for the *posterior* distribution over the latent values $\mathbf{f}$

$$\mathbb{P}(\mathbf{f}|\mathbf{y}, \mathbf{X}, \boldsymbol{\theta}) = \frac{\mathbb{P}(\mathbf{y}|\mathbf{f})\,\mathbb{P}(\mathbf{f}|\mathbf{X}, \boldsymbol{\theta})}{\int \mathbb{P}(\mathbf{y}|\mathbf{f})\,\mathbb{P}(\mathbf{f}|\mathbf{X}, \boldsymbol{\theta})\,\mathrm{d}\mathbf{f}} = \frac{\mathcal{N}(\mathbf{f}|\mathbf{0}, \mathbf{K})}{\mathbb{P}(\mathbf{y}|\mathbf{X}, \boldsymbol{\theta})} \prod_{i=1}^{n} \mathrm{sig}(y_i f_i), \tag{4.4}$$

where $Z = \mathbb{P}(\mathbf{y}|\mathbf{X}, \boldsymbol{\theta}) = \int \mathbb{P}(\mathbf{y}|\mathbf{f})\,\mathbb{P}(\mathbf{f}|\mathbf{X}, \boldsymbol{\theta})\,\mathrm{d}\mathbf{f}$ denotes the *marginal likelihood* or *evidence* for the hyperparameter $\boldsymbol{\theta}$. The joint prior over training and test latent values $\mathbf{f}$ and $\mathbf{f}_*$ given the corresponding inputs is

$$\mathbb{P}(\mathbf{f}_*, \mathbf{f}|\mathbf{X}_*, \mathbf{X}, \boldsymbol{\theta}) = \mathcal{N}\left(\begin{bmatrix} \mathbf{f} \\ \mathbf{f}_* \end{bmatrix} \middle| \mathbf{0}, \begin{bmatrix} \mathbf{K} & \mathbf{K}_* \\ \mathbf{K}_*^\top & \mathbf{K}_{**} \end{bmatrix}\right). \tag{4.5}$$

When making predictions, we marginalise over the training set latent variables

$$\mathbb{P}(\mathbf{f}_*|\mathbf{X}_*, \mathbf{y}, \mathbf{X}, \boldsymbol{\theta}) = \int \mathbb{P}(\mathbf{f}_*, \mathbf{f}|\mathbf{X}_*, \mathbf{y}, \mathbf{X}, \boldsymbol{\theta})\,\mathrm{d}\mathbf{f} = \int \mathbb{P}(\mathbf{f}_*|\mathbf{f}, \mathbf{X}_*, \mathbf{X}, \boldsymbol{\theta})\,\mathbb{P}(\mathbf{f}|\mathbf{y}, \mathbf{X}, \boldsymbol{\theta})\,\mathrm{d}\mathbf{f}, \tag{4.6}$$

where the joint posterior is factored into the product of the posterior and the conditional prior

$$\mathbb{P}\left(\mathbf{f}_*|\mathbf{f},\mathbf{X}_*,\mathbf{X},\boldsymbol{\theta}\right) \quad = \quad \mathcal{N}\left(\mathbf{f}_*|\mathbf{K}_*^\top\mathbf{K}^{-1}\mathbf{f},\mathbf{K}_{**}-\mathbf{K}_*^\top\mathbf{K}^{-1}\mathbf{K}_*\right). \tag{4.7}$$

Finally, the predictive class membership probability $p_* := \mathbb{P}\left(y_* = 1|\mathbf{x}_*,\mathbf{y},\mathbf{X},\boldsymbol{\theta}\right)$ is obtained by averaging out the test set latent variables

$$\mathbb{P}\left(y_*|\mathbf{x}_*,\mathbf{y},\mathbf{X},\boldsymbol{\theta}\right) = \int \mathbb{P}\left(y_*|f_*\right)\mathbb{P}\left(f_*|\mathbf{x}_*,\mathbf{y},\mathbf{X},\boldsymbol{\theta}\right) df_* \;=\; \int \mathrm{sig}\left(y_*f_*\right)\mathbb{P}\left(f_*|\mathbf{x}_*,\mathbf{y},\mathbf{X},\boldsymbol{\theta}\right)\mathrm{d}f_*. \tag{4.8}$$

The integral is analytically tractable for $\mathrm{sig}_{\mathrm{probit}}$ [Rasmussen and Williams, 2006, ch. 3.9] and can be efficiently approximated for $\mathrm{sig}_{\mathrm{logit}}$ [Williams and Barber, 1998, app. A].

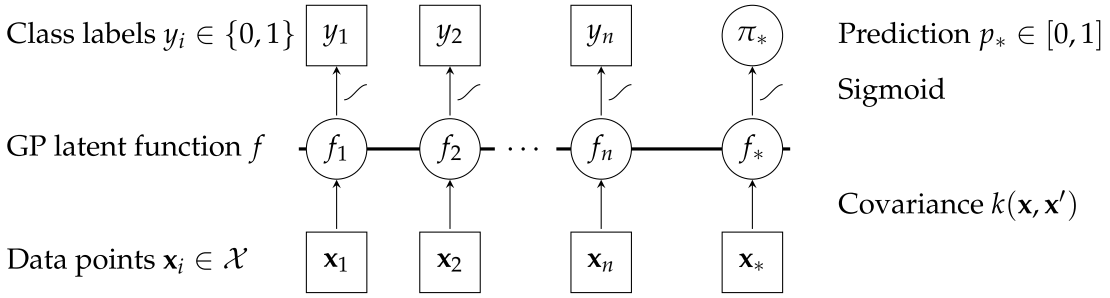

Class labels $y_i \in \{0,1\}$
y_1
y_2
y_n
π_*
Prediction $p_* \in [0,1]$
Sigmoid
GP latent function f
f_1
f_2
$\cdots$
f_n
f_*
Covariance $k(\mathbf{x},\mathbf{x}')$
Data points $\mathbf{x}_i \in \mathcal{X}$
$\mathbf{x}_1$
$\mathbf{x}_2$
$\mathbf{x}_n$
$\mathbf{x}_*$

Figure 4.1: Graphical model for binary Gaussian process classification
Circles represent unknown quantities, squares refer to observed variables. The horizontal thick line means fully connected latent variables. An observed label y_i is conditionally independent of all other nodes given the corresponding latent variable f_i. Labels y_i and latent function values f_i are connected through the sigmoid likelihood; all latent function values f_i are fully connected, since they are drawn from the same GP. The labels y_i are binary, whereas the prediction p_* is a probability and can thus have values from the whole interval $[0,1]$.

Stationary covariance functions

In preparation for the analysis of the approximation schemes described, we investigate some simple properties of the posterior for stationary covariance functions in different regimes encountered in classification. Stationary covariances of the form $k(\mathbf{x},\mathbf{x}',\boldsymbol{\theta}) = \sigma_f^2 g(|\mathbf{x}-\mathbf{x}'|/\ell)$ with $g : \mathbb{R} \to \mathbb{R}$ a monotonously decreasing function[1] and $\boldsymbol{\theta} = \{\sigma_f, \ell\}$ are widely used. The following section supplies a geometric intuition of that specific prior in the classification scenario by analysing the limiting behaviour of the covariance matrix $\mathbf{K}$ as a function of the length scale ℓ and the limiting behaviour of the likelihood as a function of the latent function scale σ_f. A pictorial illustration of the setting is given in figure 4.3.

4.2.0.1 Length scale

Two limiting cases of "ignorance with respect to the data" with marginal likelihood $Z = 2^{-n}$ can be distinguished, where $\mathbf{1} = [1,\dots 1]^\top$ and $\mathbf{I}$ is the identity matrix (see appendix F.4):

$$\begin{aligned}\lim_{\ell\to 0}\mathbf{K} &= \sigma_f^2\mathbf{I}\\ \lim_{\ell\to\infty}\mathbf{K} &= \sigma_f^2\mathbf{1}\mathbf{1}^\top.\end{aligned}$$

For very small length scales ($\ell \to 0$), the prior is simply isotropic as all points are deemed to be far away from each other and the whole model factorises. Thus, the (identical) posterior moments can be calculated dimension-wise. (See figure 4.3, regimes 1, 4 and 7.)

[1] Furthermore, we require $g(0) = 1$ and $\lim_{t\to\infty} g(t) = 0$.

For very long length scales ($\ell \to \infty$), the prior becomes degenerate as all data points are deemed to be close to each other and takes the form of a cigar along the hyper-diagonal. (See figure 4.3, regimes 3, 6 and 9.) A 1d example of functions drawn from GP priors with different lengthscales ℓ is shown in figure 4.2 on the left. The length scale has to be suited to the data; if chosen too small, we will overfit, if chosen too high underfitting will occur.

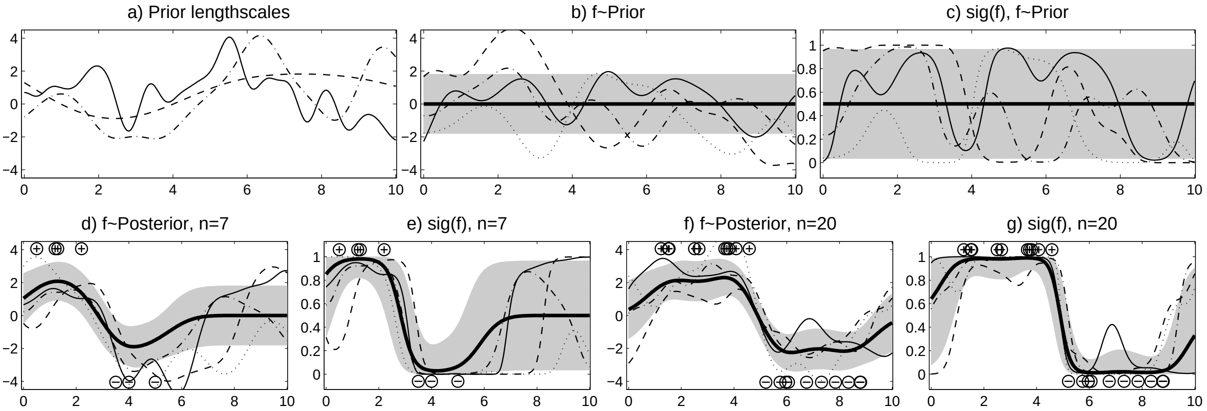


Figure 4.2: Pictorial one-dimensional illustration of binary Gaussian process classification. Plot a) shows 3 sample functions drawn from GPs with different length scales ℓ. Then, three pairs of plots show distributions over functions $f : \mathbb{R} \to \mathbb{R}$ and sig$(f) : \mathbb{R} \to [0, 1]$ occurring in GP classification. b+c) the prior, d+e) a posterior with $n = 7$ observations and f+g) a posterior with $n = 20$ observations along with the n observations with binary labels. The thick black line is the mean, the grey background is the ± standard deviation and the thin lines are sample functions. With more and more data points observed, the uncertainty is gradually shrunk. At the decision boundary the uncertainty is smallest.

4.2.0.2 Latent function scale

The sigmoid likelihood function sig $(y_i f_i)$ measures the agreement of the signs of the latent function and the label in a smooth way, i.e. values close to one if the signs of y_i and f_i are the same and $|f_i|$ is large, and values close to zero if the signs are different and $|f_i|$ is large. The latent function scale σ_f of the data can be moved into the likelihood $\tilde{\text{sig}}_{\sigma_f}(t) = \text{sig}(\sigma_f^2 t)$, thus σ_f models the steepness of the likelihood and finally the smoothness of the agreement by interpolation between the two limiting cases "ignorant" and "hard cut":

$$\begin{aligned}\lim_{\sigma_f \to 0} \text{sig}(t) &\equiv \frac{1}{2} \quad \text{“ignorant"} \\ \lim_{\sigma_f \to \infty} \text{sig}(t) &\equiv \text{step}(t) := \left\{\ 0,\ t < 0;\ \ \tfrac{1}{2},\ t = 0;\ \ 1,\ 0 < t \right. \qquad \text{“hard cut"}\end{aligned}$$

In the case of very small latent scales ($\sigma_f \to 0$), the likelihood is flat causing the posterior to equal the prior. The marginal likelihood is again $Z = 2^{-n}$. (See figure 4.3, regimes 7, 8 and 9.)

In the case of large latent scales ($\sigma_f \gg 1$), the likelihood approaches the step function. (See figure 4.3, regimes 1, 2 and 3.) A further increase of the latent scale does not change the model anymore. The model is effectively the same for all σ_f above a threshold.

4.2.1 Gaussian approximations

Unfortunately, the posterior over the latent values (equation 4.4) is not Gaussian due to the non-Gaussian likelihood (equation 4.3). Therefore, the latent distribution (equation 4.6), the predictive distribution (equation 4.8) and the marginal likelihood Z cannot be written as analytical

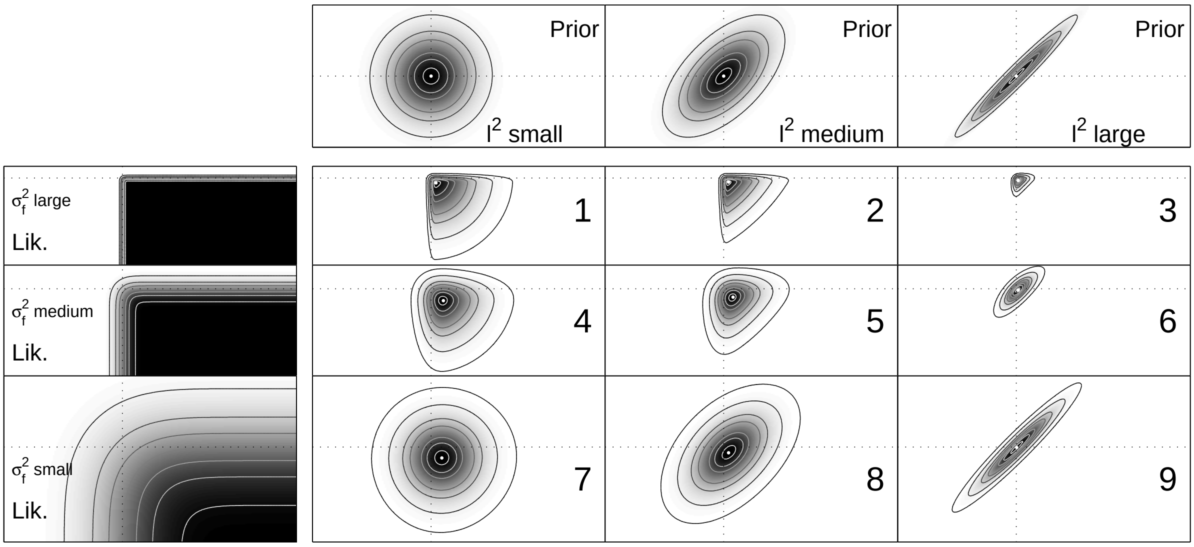


Figure 4.3: Gaussian process classification: prior, likelihood and exact posterior. Nine numbered quadrants show the posterior obtained by multiplication of different priors and likelihoods. The leftmost column illustrates the likelihood function for three different steepness parameters σ_f and the upper row depicts the prior for three different length scales ℓ. Here, we use σ_f as a parameter of the likelihood. Alternatively, rows correspond to "degree of Gaussianity" and columns stand for "degree of isotropy". The axes show the latent function values $f_1 = f(\mathbf{x}_1)$ and $f_2 = f(\mathbf{x}_2)$. A simple toy example employing the cumulative Gaussian likelihood and a squared exponential covariance $k(\mathbf{x}, \mathbf{x}') = \sigma_f^2 \exp(-\|\mathbf{x} - \mathbf{x}'\|^2 / 2\ell^2)$ with length scales $\ln \ell = \{0, 1, 2.5\}$ and latent function scales $\ln \sigma_f = \{-1.5, 0, 1.5\}$ is used. Two data points $\mathbf{x}_1 = \sqrt{2}$, $\mathbf{x}_2 = -\sqrt{2}$ with corresponding labels $y_1 = 1$, $y_2 = -1$ form the dataset.

expressions[2]. To obtain exact answers, one can resort to sampling algorithms (MCMC). However, if sig is concave in the logarithmic domain, the posterior can be shown to be unimodal motivating Gaussian approximations to the posterior. Five different Gaussian approximations corresponding to methods explained later onwards are depicted in figure 4.4.

A quadratic approximation to the log likelihood $\phi(f_i) := \ln \mathbb{P}(y_i|f_i)$ at $\tilde{f}_i$

$$\phi(f_i) \quad \approx \quad \phi(\tilde{f}_i) + \phi'(\tilde{f}_i)(f_i - \tilde{f}_i) + \frac{1}{2}\phi''(\tilde{f}_i)(f_i - \tilde{f}_i)^2 \;=\; -\frac{1}{2}w_i f_i^2 + b_i f_i + \text{const}_{f_i}$$

motivates the following approximate posterior $\mathbb{Q}(\mathbf{f}|\mathbf{y}, \mathbf{X}, \boldsymbol{\theta})$

$$\begin{aligned}
\ln \mathbb{P}(\mathbf{f}|\mathbf{y}, \mathbf{X}, \boldsymbol{\theta}) \quad & \overset{(4.4)}{=} \quad -\frac{1}{2}\mathbf{f}^\top \mathbf{K}^{-1}\mathbf{f} + \sum_{i=1}^{n} \ln \mathbb{P}(y_i|f_i) + \text{const}_{\mathbf{f}} \\
& \overset{\text{quad. approx.}}{\approx} \quad -\frac{1}{2}\mathbf{f}^\top \mathbf{K}^{-1}\mathbf{f} - \frac{1}{2}\mathbf{f}^\top \mathbf{W}\mathbf{f} + \mathbf{b}^\top \mathbf{f} + \text{const}_{\mathbf{f}} \\
& \overset{\mathbf{m}:=(\mathbf{K}^{-1}+\mathbf{W})^{-1}\mathbf{b}}{=} \quad -\frac{1}{2}(\mathbf{f} - \mathbf{m})^\top \left(\mathbf{K}^{-1} + \mathbf{W}\right)(\mathbf{f} - \mathbf{m}) + \text{const}_{\mathbf{f}} \\
& = \quad \ln \mathcal{N}(\mathbf{f}|\mathbf{m}, \mathbf{V}) \;=:\; \ln \mathbb{Q}(\mathbf{f}|\mathbf{y}, \mathbf{X}, \boldsymbol{\theta}), \qquad (4.9)
\end{aligned}$$

where $\mathbf{V}^{-1} = \mathbf{K}^{-1} + \mathbf{W}$ and $\mathbf{W}$ denotes the precision of the effective likelihood (see equation

[2] One can write down exact expressions for the first two moments $m_*(\mathbf{x})$ and $k_*(\mathbf{x}, \mathbf{x}')$ of the posterior process $f_*(\mathbf{x})$ conditioned on the observed data $\mathcal{D} = (\mathbf{y}, \mathbf{X})$ but the involved integrals are not tractable[Csató and Opper, 2002]:

$$\begin{array}{lllll}
m_*(\mathbf{x}) & = \mathbb{E}\left[f_*(\mathbf{x})|\mathcal{D}\right] & = \mathbf{k}_*^\top \boldsymbol{\alpha} & \quad & \boldsymbol{\alpha} = \frac{1}{Z}\int \mathbb{P}(\mathbf{f}|\mathbf{X}, \boldsymbol{\theta}) \frac{\partial \mathbb{P}(\mathbf{y}|\mathbf{f})}{\partial \mathbf{f}} d\mathbf{f} \\
k_*(\mathbf{x}, \mathbf{x}') & = \mathbb{C}\left[f_*(\mathbf{x}), f_*(\mathbf{x}')|\mathcal{D}\right] & = k_{**} + \mathbf{k}_*^\top \mathbf{C}^{-1}\mathbf{k}_*' & \quad & \mathbf{C}^{-1} = \frac{1}{Z}\int \mathbb{P}(\mathbf{f}|\mathbf{X}, \boldsymbol{\theta}) \frac{\partial^2 \mathbb{P}(\mathbf{y}|\mathbf{f})}{\partial \mathbf{f} \partial \mathbf{f}^\top} d\mathbf{f} - \boldsymbol{\alpha}\boldsymbol{\alpha}^\top
\end{array}$$

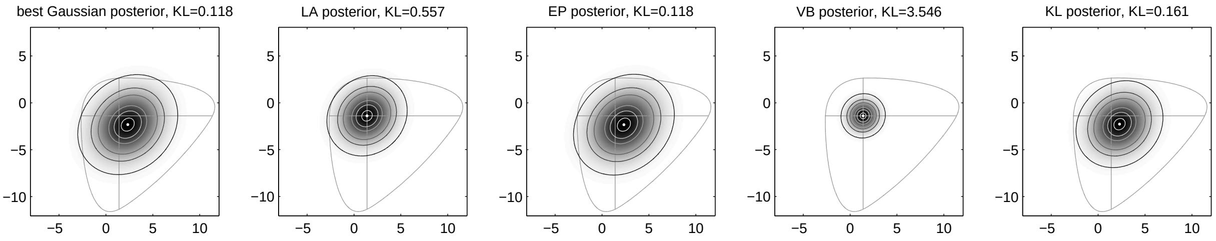


Figure 4.4: Five Gaussian approximations to the posterior
Different Gaussian approximations to the exact posterior (in grey) using the regime 2 setting of figure 4.3 are shown. The exact posterior is represented in grey by a cross at the mode and a single equiprobability contour line. From left to right: the best Gaussian approximation (intractable) matches the moments of the true posterior, the Laplace approximation does a Taylor expansion around the mode, the EP approximation iteratively matches marginal moments, the variational method maximises a lower bound on the marginal likelihood and the KL method minimises the Kullback-Leibler to the exact posterior. The axes show the latent function values $f_1 = f(\mathbf{x}_1)$ *and* $f_2 = f(\mathbf{x}_2)$.

4.11). It turns out that the methods discussed in the following sections correspond to particular choices of $\mathbf{m}$ and $\mathbf{V}$.

Let us assume, we found such a Gaussian approximation to the posterior with mean $\mathbf{m}$ and (co)variance $\mathbf{V}$. Consequently, the latent distribution for a test point becomes a tractable one-dimensional Gaussian $\mathbb{P}(f_*|\mathbf{x}_*, \mathbf{y}, \mathbf{X}, \boldsymbol{\theta}) = \mathcal{N}(f_*|\mu_*, \sigma_*^2)$ with the following moments [Rasmussen and Williams, 2006, p. 44 and 56]:

$$\begin{array}{rclcrcl} \mu_* & = & \mathbf{k}_*^\top \mathbf{K}^{-1}\mathbf{m} & = & \mathbf{k}_*^\top \boldsymbol{\alpha} & & \boldsymbol{\alpha} = \mathbf{K}^{-1}\mathbf{m} \\ \sigma_*^2 & = & k_{**} - \mathbf{k}_*^\top \left(\mathbf{K}^{-1} - \mathbf{K}^{-1}\mathbf{V}\mathbf{K}^{-1}\right)\mathbf{k}_* & = & k_{**} - \mathbf{k}_*^\top \left(\mathbf{K} + \mathbf{W}^{-1}\right)^{-1}\mathbf{k}_* & & \end{array} \quad (4.10)$$

Since Gaussians are closed under multiplication, one can – given the Gaussian prior $\mathbb{P}(\mathbf{f}|\mathbf{X}, \boldsymbol{\theta})$ and the Gaussian approximation to the posterior $\mathbb{Q}(\mathbf{f}|\mathbf{y}, \mathbf{X}, \boldsymbol{\theta})$ – deduce the Gaussian factor $\mathbb{Q}(\mathbf{y}|\mathbf{f})$ so that $\mathbb{Q}(\mathbf{f}|\mathbf{y}, \mathbf{X}, \boldsymbol{\theta}) \propto \mathbb{Q}(\mathbf{y}|\mathbf{f})\,\mathbb{P}(\mathbf{f}|\mathbf{X}, \boldsymbol{\theta})$. Consequently, this Gaussian factor can be thought of as an *effective likelihood*. Five different effective likelihoods, corresponding to methods discussed subsequently, are depicted in figure 4.5. By "dividing" the approximate Gaussian posterior (see appendix F.5) by the true Gaussian prior we find the contribution of the effective likelihood $\mathbb{Q}(\mathbf{y}|\mathbf{f})$:

$$\mathbb{Q}(\mathbf{y}|\mathbf{f}) \propto \frac{\mathcal{N}(\mathbf{f}|\mathbf{m}, \mathbf{V})}{\mathcal{N}(\mathbf{f}|\mathbf{0}, \mathbf{K})} \propto \mathcal{N}\left(\mathbf{f}|\,(\mathbf{K}\mathbf{W})^{-1}\mathbf{m} + \mathbf{m}, \mathbf{W}^{-1}\right) \quad (4.11)$$

We see (also from equation 4.9) that $\mathbf{W}$ models the precision of the effective likelihood. In general, $\mathbf{W}$ is a full matrix containing n^2 parameters.[3] However, all algorithms maintaining a Gaussian posterior approximation work with a diagonal $\mathbf{W}$ to enforce the effective likelihood to factorise over examples (as the true likelihood does, see figure 4.1) in order to reduce the number of parameters. We are not aware of work quantifying the error made by this assumption.

4.2.2 Sparse approximations

Different authors proposed to sparsify Gaussian process classification to achieve computational tractability. The support vector machine is naturally a sparse kernel machine, however it cannot

[3] A non-diagonal matrix $\mathbf{W} = \begin{bmatrix} 1.4834 & -0.4500 \\ -0.4500 & 1.4834 \end{bmatrix}$ is obtained from $\mathbf{K} = \begin{bmatrix} 1 & 0.9 \\ 0.9 & 1 \end{bmatrix}$, $y_1 = y_2 = 1$ and step function likelihood $\mathbb{P}(y_i|f_i) = (\text{sign}(y_i f_i) + 1)/2$ by numerical moment matching on a grid with $n = 1000$ on the interval $f_i \in [-5, 5]$ $\mathbf{m} = \begin{bmatrix} 0.8850 \\ 0.8850 \end{bmatrix}$, $\mathbf{V} = \begin{bmatrix} 0.3625 & 0.2787 \\ 0.2787 & 0.3625 \end{bmatrix}$.

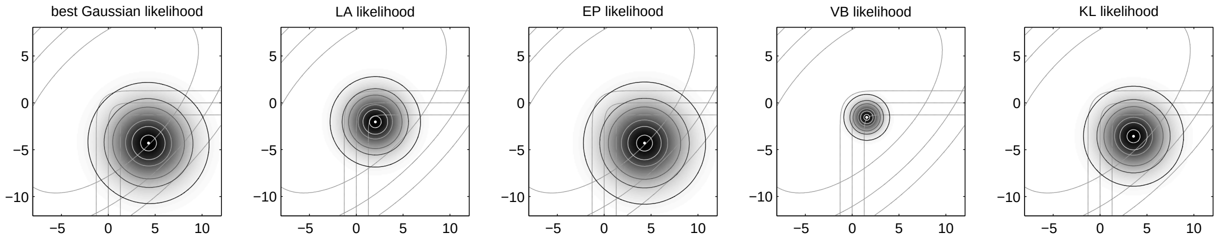


Figure 4.5: Five effective likelihoods

A Gaussian approximation to the posterior induces a Gaussian effective likelihood (equation 4.11). Exact prior and likelihood are shown in grey. Different effective likelihoods are shown; order and setting are the same as described in figure 4.4. The axes show the latent function values $f_1 = f(\mathbf{x}_1)$ and $f_2 = f(\mathbf{x}_2)$. The effective likelihood replaces the non-Gaussian likelihood (indicated by three grey lines). A good replacement behaves like the exact likelihood in regions of high prior density (indicated by grey ellipses). EP and KL yield a good coverage of that region. However LA and VB yield too concentrated replacements.

entirely be interpreted in a probabilistic framework [Sollich, 2002]. Sparse online Gaussian processes (SOGP) were derived in Csató [2002], the informative vector machine (IVM) was introduced by [Lawrence et al., 2004] and the relevance vector machine (RVM) was suggested by Tipping [2001]. SOGP keep an active set of expansion vectors, discarded data points are represented as a projection in the subspace of the active set. The IVM is a method for greedily forward selecting informative data-points based on information theoretic measures. The RVM is a degenerate Gaussian process that does not lead to reliable posterior variance estimates [Rasmussen and Quiñonero-Candela, 2005].

4.2.3 Marginal likelihood

Prior knowledge over the latent function f is encoded in the choice of a covariance function k containing hyperparameters $\boldsymbol{\theta}$. In principle, one can do inference jointly over f and $\boldsymbol{\theta}$, e.g. by sampling techniques. Another approach to model selection is maximum likelihood type II also known as the evidence framework of MacKay [1992], where the hyperparameters $\boldsymbol{\theta}$ are chosen to maximise the marginal likelihood or evidence $\mathbb{P}(\mathbf{y}|\mathbf{X},\boldsymbol{\theta})$. In other words, one maximises the agreement between observed data and the model. Therefore, one has a strong motivation to estimate the marginal likelihood.

Geometrically, the marginal likelihood measures the volume of the prior times the likelihood. High volume implies a strong consensus between our initial belief and our observations. In GP classification, each data point $\mathbf{x}_i$ gives rise to a dimension f_i in latent space. The likelihood implements a mechanism, for smoothly restricting the posterior along the axis of f_i to the side corresponding to the sign of y_i . Thus, the latent space $\mathbb{R}^n$ is softly cut down to the orthant given by the values in $\mathbf{y}$. The log marginal likelihood measures, what fraction of the prior lies in that orthant. Finally, the value $Z = 2^{-n}$ corresponds to the case, where half of the prior lies on either side along each axis in latent space. Consequently, successful inference is characterised by $Z > 2^{-n}$.

Some posterior approximations (sections 4.3 and 4.4) also provide an approximation to the marginal likelihood, other methods provide a lower bound (sections 4.5 and 4.6). Any Gaussian approximation $\mathbb{Q}(\mathbf{f}|\boldsymbol{\theta}) = \mathcal{N}(\mathbf{f}|\mathbf{m},\mathbf{V})$ to the posterior $\mathbb{P}(\mathbf{f}|\mathbf{y},\mathbf{X},\boldsymbol{\theta})$ gives rise to a lower bound Z_B to the marginal likelihood Z by application of Jensen's inequality. This bound is also used in the context of sparse approximations [Seeger, 2003].

$$\begin{aligned}\ln Z = \ln \mathbb{P}(\mathbf{y}|\mathbf{X},\boldsymbol{\theta}) &= \ln \int \mathbb{P}(\mathbf{y}|\mathbf{f})\,\mathbb{P}(\mathbf{f}|\mathbf{X},\boldsymbol{\theta})\,\mathrm{d}\mathbf{f} = \ln \int \mathbb{Q}(\mathbf{f}|\boldsymbol{\theta})\,\frac{\mathbb{P}(\mathbf{y}|\mathbf{f})\,\mathbb{P}(\mathbf{f}|\mathbf{X},\boldsymbol{\theta})}{\mathbb{Q}(\mathbf{f}|\boldsymbol{\theta})}\mathrm{d}\mathbf{f} \\ &\overset{\text{Jensen}}{\geq} \int \mathbb{Q}(\mathbf{f}|\boldsymbol{\theta}) \ln \frac{\mathbb{P}(\mathbf{y}|\mathbf{f})\,\mathbb{P}(\mathbf{f}|\mathbf{X},\boldsymbol{\theta})}{\mathbb{Q}(\mathbf{f}|\boldsymbol{\theta})}\mathrm{d}\mathbf{f} =: \ln Z_{KL} \qquad (4.12)\end{aligned}$$

Some algebra (appendix F.6) leads to the following expression for $\ln Z_{KL}$:

$$\underbrace{\sum_{i=1}^{n}\int\mathcal{N}(f|,0,1)\ln\operatorname{sig}\left(y_i\left\{\sqrt{V_{ii}}f+m_i\right\}\right)\mathrm{d}f}_{\text{1) data fit}}+\frac{1}{2}[n-\underbrace{\mathbf{m}^\top\mathbf{K}^{-1}\mathbf{m}}_{\text{2) data fit}}+\underbrace{\ln\left|\mathbf{V}\mathbf{K}^{-1}\right|-\operatorname{tr}\left(\mathbf{V}\mathbf{K}^{-1}\right)}_{\text{3) regularizer}}] \tag{4.13}$$

Model selection means maximisation of $\ln Z_{KL}$. Term 1) is a sum of one-dimensional Gaussian integrals of sigmoid functions in the logarithmic domain with adjustable offset and steepness. The integrals can be numerically computed in an efficient way using Gauss-Hermite quadrature [Press et al., 1993, §4.5]. As the sigmoid in the log domain takes only negative values, the first term will be negative. That means, maximisation of the first term is done by shifting the log-sigmoid so that the high-density region of the Gaussian is multiplied by small values. Term 2) is the equivalent of the data-fit term in GP regression [Rasmussen and Williams, 2006, ch. 5.4.1]. Thus, the first and the second term encourage fitting the data by favouring small variances V_{ii} and large means m_i having the same sign as y_i. The third term can be rewritten as $-\ln|\mathbf{I}+\mathbf{KW}|-\operatorname{tr}\left((\mathbf{I}+\mathbf{KW})^{-1}\right)$ and yields $-\sum_{i=1}^{n}\ln(1+\lambda_i)+\frac{1}{1+\lambda_i}$ with $\lambda_i\geq 0$ being the eigenvalues of $\mathbf{KW}$. Thus, term 3) keeps the eigenvalues of $\mathbf{KW}$ small, thereby favouring a smaller class of functions – this can be seen as an instance of Occam's razor.

Furthermore, the bound

$$\ln Z_{KL} \;=\; \int\mathbb{Q}(\mathbf{f}|\boldsymbol{\theta})\ln\frac{\mathbb{P}(\mathbf{f}|\mathbf{y},\mathbf{X},\boldsymbol{\theta})\,\mathbb{P}(\mathbf{y}|\mathbf{X})}{\mathbb{Q}(\mathbf{f}|\boldsymbol{\theta})}\mathrm{d}\mathbf{f} \;=\; \ln Z-\mathrm{KL}\left(\mathbb{Q}(\mathbf{f}|\boldsymbol{\theta})\;\|\;\mathbb{P}(\mathbf{f}|\mathbf{y},\mathbf{X},\boldsymbol{\theta})\right) \tag{4.14}$$

can be decomposed into the exact marginal likelihood minus the Kullback-Leibler (KL) divergence between the exact posterior and the approximate posterior. Thus by maximising the lower bound $\ln Z_{KL}$ on $\ln Z$, we effectively minimise the KL-divergence between $\mathbb{P}(\mathbf{f}|\mathbf{y},\mathbf{X},\boldsymbol{\theta})$ and $\mathbb{Q}(\mathbf{f}|\boldsymbol{\theta})=\mathcal{N}(\mathbf{f}|\mathbf{m},\mathbf{V})$. The bound is tight if and only if $\mathbb{Q}(\mathbf{f}|\boldsymbol{\theta})=\mathbb{P}(\mathbf{f}|\mathbf{y},\mathbf{X},\boldsymbol{\theta})$.

4.3 Laplace's method (LA)

A second order Taylor expansion around the posterior mode $\mathbf{m}$ leads to a natural way of constructing a Gaussian approximation to the log-posterior $\boldsymbol{\Psi}(\mathbf{f})=\ln\mathbb{P}(\mathbf{f}|\mathbf{y},\mathbf{X},\boldsymbol{\theta})$ [Williams and Barber, 1998, Rasmussen and Williams, 2006, ch. 3]. The mode $\mathbf{m}$ is taken as the mean of the approximate Gaussian. Linear terms of $\boldsymbol{\Psi}$ vanish because the gradient at the mode is zero. The quadratic term of $\boldsymbol{\Psi}$ is given by the negative Hessian $\mathbf{W}$, which - due to the likelihood's factorial structure - turns out to be diagonal. The mode $\mathbf{m}$ is found by Newton's method.

Posterior

$$\begin{aligned}
\mathbb{P}(\mathbf{f}|\mathbf{y},\mathbf{X},\boldsymbol{\theta}) &\approx \mathcal{N}(\mathbf{f}|\mathbf{m},\mathbf{V})=\mathcal{N}\left(\mathbf{f}|\mathbf{m},\left(\mathbf{K}^{-1}+\mathbf{W}\right)^{-1}\right)\\
\mathbf{m} &= \arg\max_{\mathbf{f}\in\mathbb{R}^n}\mathbb{P}(\mathbf{y}|\mathbf{f})\,\mathbb{P}(\mathbf{f}|\mathbf{X},\boldsymbol{\theta})\\
\mathbf{W} &= -\left.\frac{\partial^2\ln\mathbb{P}(\mathbf{y}|\mathbf{f})}{\partial\mathbf{f}\partial\mathbf{f}^\top}\right|_{\mathbf{f}=\mathbf{m}}=-\left[\left.\frac{\partial^2\ln\mathbb{P}(y_i|f_i)}{\partial f_i^2}\right|_{f_i=m_i}\right]_{ii}
\end{aligned}$$

Marginal likelihood

The unnormalised posterior $\mathbb{P}(\mathbf{y}|\mathbf{f})\,\mathbb{P}(\mathbf{f}|\mathbf{X},\boldsymbol{\theta})$ has its maximum $h=\exp(\boldsymbol{\Psi}(\mathbf{m}))$ at its mode $\mathbf{m}$, where the gradient vanishes. A Taylor expansion of $\boldsymbol{\Psi}$ is then given by $\boldsymbol{\Psi}(\mathbf{f})\approx h-\frac{1}{2}(\mathbf{f}-\mathbf{m})^\top(\mathbf{K}^{-1}+\mathbf{W})(\mathbf{f}-\mathbf{m})$. Consequently, the log marginal likelihood can be approximated by plugging in the approximation of $\boldsymbol{\Psi}(\mathbf{f})$.

$$\begin{aligned}\ln Z = \ln \mathbb{P}(\mathbf{y}|\mathbf{X},\boldsymbol{\theta}) &= \ln \int \mathbb{P}(\mathbf{y}|\mathbf{f})\,\mathbb{P}(\mathbf{f}|\mathbf{X},\boldsymbol{\theta})\,d\mathbf{f} = \ln \int \exp(\boldsymbol{\Psi}(\mathbf{f}))\,d\mathbf{f} \\ &\approx \ln h + \ln \int \exp\left(-\frac{1}{2}(\mathbf{f}-\mathbf{m})^\top \left(\mathbf{K}^{-1}+\mathbf{W}\right)(\mathbf{f}-\mathbf{m})\right) d\mathbf{f} \\ &= -\ln \mathbb{P}(\mathbf{y}|\mathbf{m}) - \frac{1}{2}\mathbf{m}^\top \mathbf{K}^{-1}\mathbf{m} - \frac{1}{2}\ln|\mathbf{I}+\mathbf{KW}| \qquad (4.15)\end{aligned}$$

4.4 Expectation propagation (EP)

EP [Minka, 2001b] is an iterative method to find approximations based on approximate marginal moments, which can be applied to Gaussian processes. See Rasmussen and Williams [2006, ch. 3] for details. The individual likelihood terms are replaced by unnormalised Gaussians

$$\mathbb{P}(y_i|f_i) \approx Z_i^{-1}\mathcal{N}(f_i|\mu_i,\sigma_i^2)$$

so that the approximate marginal moments of $\mathbb{Q}(f_i) := \int \mathcal{N}(\mathbf{f}|\mathbf{0},\mathbf{K}) \prod_{j=1}^n Z_j^{-1}\mathcal{N}\left(f_j|\mu_j,\sigma_j^2\right) d\mathbf{f}_{\neg i}$ agree with the marginals of $\int \mathcal{N}(\mathbf{f}|\mathbf{0},\mathbf{K})\mathbb{P}(y_i|f_i)\prod_{j\neq i} Z_j^{-1}\mathcal{N}\left(f_j|\mu_j,\sigma_j^2\right) d\mathbf{f}_{\neg i}$ of the approximation based on the exact likelihood term $\mathbb{P}(y_j|f_j)$. That means, there are $3n$ quantities μ_i, σ_i^2 and Z_i to be iteratively optimised. Convergence of EP is not generally guaranteed, but there always exists a fixed-point for the EP updates in GP classification [Minka, 2001a]. If the EP iterations converge, the solution obtained is a saddle point of a special energy function [Minka, 2001a]. However, an EP update does not necessarily imply a decrease in energy. For our case of log-concave likelihood functions, we always observed convergence, but we are not aware of a formal proof.

Posterior

Based on these local approximations, the approximate posterior can be written as

$$\begin{aligned}\mathbb{P}(\mathbf{f}|\mathbf{y},\mathbf{X},\boldsymbol{\theta}) &\approx \mathcal{N}(\mathbf{f}|\mathbf{m},\mathbf{V}) = \mathcal{N}\left(\mathbf{f}|\mathbf{m},\left(\mathbf{K}^{-1}+\mathbf{W}\right)^{-1}\right) \\ \mathbf{W} &= [\sigma_i^{-2}]_{ii} \\ \mathbf{m} &= \mathbf{VW}\boldsymbol{\mu} = \left[\mathbf{I}-\mathbf{K}\left(\mathbf{K}+\mathbf{W}^{-1}\right)^{-1}\right]\mathbf{KW}\boldsymbol{\mu},\ \boldsymbol{\mu} = (\mu_1,\ldots,\mu_n)^\top\end{aligned}$$

Marginal likelihood

From the likelihood approximations, one can directly obtain an expression for the approximate log marginal likelihood.

$$\begin{aligned}\ln Z = \ln \mathbb{P}(\mathbf{y}|\mathbf{X},\boldsymbol{\theta}) &= \ln \int \mathbb{P}(\mathbf{y}|\mathbf{f})\,\mathbb{P}(\mathbf{f}|\mathbf{X},\boldsymbol{\theta})\,d\mathbf{f} \\ &\approx \ln \int \prod_{i=1}^n t\left(f_i,\mu_i,\sigma_i^2,Z_i\right)\mathbb{P}(\mathbf{f}|\mathbf{X},\boldsymbol{\theta})\,d\mathbf{f} \\ &= -\sum_{i=1}^n \ln Z_i - \frac{1}{2}\boldsymbol{\mu}^\top\left(\mathbf{K}+\mathbf{W}^{-1}\right)^{-1}\boldsymbol{\mu} - \frac{1}{2}\ln\left|\mathbf{K}+\mathbf{W}^{-1}\right| - \frac{n}{2}\ln 2\pi \qquad (4.16) \\ &= -\sum_{i=1}^n \ln\frac{Z_i}{\sqrt{2\pi}} - \frac{1}{2}\mathbf{m}^\top\left(\mathbf{K}^{-1}+\mathbf{K}^{-1}\mathbf{W}^{-1}\mathbf{K}^{-1}\right)\mathbf{m} - \frac{1}{2}\ln\left|\mathbf{K}+\mathbf{W}^{-1}\right| =: \ln Z_{EP}\end{aligned}$$

The lower bound provided by Jensen's inequality Z_{KL} (equation 4.13) is known to be below the approximation Z_{EP} obtained by EP [Opper and Winther, 2005, page 2183]. From $Z_{EP} \geq Z_{KL}$ and $Z \geq Z_{KL}$ it is not clear, which value one should use. In principle, Z_{EP} could be an inaccurate approximation. However, our experimental findings and extensive Monte Carlo simulations suggest that Z_{EP} is almost exact.

4.4.1 Thouless, Anderson & Palmer method (TAP)

Based on ideas rooted in Statistical Physics, one can approach the problem from a slightly different angle [Opper and Winther, 2000]. Individual Gaussian approximations $\mathcal{N}(f_i|\mu_{\neg i}, \sigma^2_{\neg i})$ are only made to predictive distributions $\mathbb{P}\left(f_i|\mathbf{x}_i, \mathbf{y}_{\setminus i}, \mathbf{X}_{\setminus i}, \boldsymbol{\theta}\right)$ for data points $\mathbf{x}_i$ that have been previously removed from the training set. Based on $\mu_{\neg i}$ and $\sigma^2_{\neg i}$ one can derive explicit expressions for $(\boldsymbol{\alpha}, \mathbf{W}^{\frac{1}{2}})$, our parameters of interest.

$$\begin{aligned} \alpha_i &\approx \frac{\int \frac{\partial}{\partial f_i}\mathbb{P}\left(y_i|f_i\right)\mathcal{N}(f_i|\mu_{\neg i}, \sigma^2_{\neg i})\mathrm{d}f_i}{\int \mathbb{P}\left(y_i|f_i\right)\mathcal{N}(f_i|\mu_{\neg i}, \sigma^2_{\neg i})\mathrm{d}f_i} \\ \left[\mathbf{W}^{-1}\right]_{ii} &\approx \sigma^2_{\neg i}\left(\frac{1}{\alpha_i\left[\mathbf{K}\boldsymbol{\alpha}\right]_i} - 1\right) \end{aligned} \tag{4.17}$$

In turn, the $2n$ parameters $(\mu_{\neg i}, \sigma^2_{\neg i})$ can be expressed as a function of $\boldsymbol{\alpha}$, $\mathbf{K}$ and $\mathbf{W}^{\frac{1}{2}}$.

$$\begin{aligned} \sigma^2_{\neg i} &= 1/\left[\left(\mathbf{K}+\mathbf{W}^{-1}\right)^{-1}\right]_{ii} - \left[\mathbf{W}^{-1}\right]_{ii} \\ \mu_{\neg i} &= \left[\mathbf{K}\boldsymbol{\alpha}\right]_i - \sigma^2_{\neg i}\alpha_i \end{aligned} \tag{4.18}$$

As a result, a system (equations 4.17/4.18) of nonlinear equations in $\mu_{\neg i}$ and $\sigma^2_{\neg i}$ has to be solved by iteration. Each step involves a matrix inversion of cubic complexity. A faster "naïve" variant updating only n parameters has also been proposed in Opper and Winther [2000] but it does not lead to the same fixed point. As in the FV algorithm (section 4.7), a formal complex transformation leads to a simplified version by fixing $\sigma^2_{\neg i} = \mathbf{K}_{ii}$, called (TAPnaive) in the sequel.

Finally, for prediction, the predictive posterior $\mathbb{P}\left(f_*|\mathbf{x}_*, \mathbf{y}, \mathbf{X}, \boldsymbol{\theta}\right)$ is approximated by a Gaussian $\mathcal{N}(f_*|\mu_*, \sigma^2_*)$ at a test point $\mathbf{x}_*$ based on the parameters $(\boldsymbol{\alpha}, \mathbf{W}^{\frac{1}{2}})$ and according to equation (4.10).

A fixed-point of the TAP mean-field equations is also a fixed-point of the EP algorithm [Minka, 2001a]. This theoretical result was confirmed in our numerical simulations. However, the EP algorithm is more practical and typically much faster. For this reason, we are not going to treat the TAP method as an independent algorithm.

4.5 KL-divergence minimisation (KL)

In principle, we simply want to minimise a dissimilarity measure between the approximate posterior $\mathbb{Q}\left(\mathbf{f}|\boldsymbol{\theta}\right) = \mathcal{N}\left(\mathbf{f}|\mathbf{m}, \mathbf{V}\right)$ and the exact posterior $\mathbb{P}\left(\mathbf{f}|\mathbf{y}, \mathbf{X}, \boldsymbol{\theta}\right)$. One quantity to minimise is the KL-divergence

$$\mathrm{KL}\left(\mathbb{P}\left(\mathbf{f}|\mathbf{y}, \mathbf{X}, \boldsymbol{\theta}\right) \,||\, \mathbb{Q}\left(\mathbf{f}|\boldsymbol{\theta}\right)\right) = \int \mathbb{P}\left(\mathbf{f}|\mathbf{y}, \mathbf{X}, \boldsymbol{\theta}\right) \ln \frac{\mathbb{P}\left(\mathbf{f}|\mathbf{y}, \mathbf{X}, \boldsymbol{\theta}\right)}{\mathbb{Q}\left(\mathbf{f}|\boldsymbol{\theta}\right)}\mathrm{d}\mathbf{f}.$$

Unfortunately, this expression is intractable. If instead, we measure the reverse KL-divergence, we regain tractability

$$\mathrm{KL}\left(\mathbb{Q}\left(\mathbf{f}|\boldsymbol{\theta}\right) \,||\, \mathbb{P}\left(\mathbf{f}|\mathbf{y}, \mathbf{X}, \boldsymbol{\theta}\right)\right) = \int \mathcal{N}\left(\mathbf{f}|\mathbf{m}, \mathbf{V}\right) \ln \frac{\mathcal{N}\left(\mathbf{f}|\mathbf{m}, \mathbf{V}\right)}{\mathbb{P}\left(\mathbf{f}|\mathbf{y}, \mathbf{X}, \boldsymbol{\theta}\right)}\mathrm{d}\mathbf{f} =: \mathrm{KL}(\mathbf{m}, \mathbf{V}).$$

A similar approach has been followed for regression with Laplace or Cauchy noise [Opper and Archambeau, 2009]. Finally, we minimise the following objective (see appendix F.6) with respect to the variables $\mathbf{m}$ and $\mathbf{V}$. Constant terms have been dropped from the expression

$$\mathrm{KL}(\mathbf{m}, \mathbf{V}) \overset{\mathrm{c}}{=} -\int \mathcal{N}(f)\left[\sum_{i=1}^{n} \ln \mathrm{sig}\left(\sqrt{v_{ii}}y_i f + m_i y_i\right)\right]\mathrm{d}f - \frac{1}{2}\ln|\mathbf{V}| + \frac{1}{2}\mathbf{m}^\top\mathbf{K}^{-1}\mathbf{m} + \frac{1}{2}\mathrm{tr}\left(\mathbf{K}^{-1}\mathbf{V}\right).$$

We refer to the first term of $\text{KL}(\mathbf{m}, \mathbf{V})$ as $a(\mathbf{m}, \mathbf{V})$ to keep the expressions short. We calculate first derivatives and equate them with zero to obtain necessary conditions that have to be fulfilled at a local optimum $(\mathbf{m}^*, \mathbf{V}^*)$

$$\begin{aligned}
\frac{\partial \text{KL}}{\partial \mathbf{m}} &= \frac{\partial a}{\partial \mathbf{m}} - \mathbf{K}^{-1}\mathbf{m} = \mathbf{0} \;\Rightarrow\; \mathbf{K}^{-1}\mathbf{m} = \frac{\partial a}{\partial \mathbf{m}} = \boldsymbol{\alpha} \\
\frac{\partial \text{KL}}{\partial \mathbf{V}} &= \frac{\partial a}{\partial \mathbf{V}} + \frac{1}{2}\mathbf{V}^{-1} - \frac{1}{2}\mathbf{K}^{-1} = \mathbf{0} \;\Rightarrow\; \mathbf{V} = \left(\mathbf{K}^{-1} - 2\frac{\partial a}{\partial \mathbf{V}}\right)^{-1} = \left(\mathbf{K}^{-1} - 2\boldsymbol{\Lambda}\right)^{-1},
\end{aligned}$$

which defines $\boldsymbol{\Lambda}$. If the approximate posterior is parametrised by $(\mathbf{m}, \mathbf{V})$, there are $\mathcal{O}(n^2)$ free variables. Once the necessary conditions for a local minimum are fulfilled (i.e. the derivatives $\partial\text{KL}/\partial\mathbf{m}$ and $\partial\text{KL}/\partial\mathbf{V}$ vanish), the problem can be re-parametrised in terms of $(\boldsymbol{\alpha}, \boldsymbol{\Lambda})$. Since $\boldsymbol{\Lambda} = \partial a/\partial\mathbf{V}$ is a diagonal matrix (see equations 2.16 and F.3), the optimum is characterised by $2n$ free parameters. This fact was pointed out by Manfred Opper (personal communication) and mentioned in Seeger [1999, ch. 5.21, eq. 5.3]. Thus, a minimisation scheme based on Newton iterations on the joint vector $\boldsymbol{\xi} := [\boldsymbol{\alpha}^\top, \boldsymbol{\Lambda}_{ii}]^\top$ takes $\mathcal{O}(8 \cdot n^3)$ operations. Details about the derivatives $\partial\text{KL}/\partial\boldsymbol{\xi}$ and $\partial^2\text{KL}/\partial\boldsymbol{\xi}\partial\boldsymbol{\xi}^\top$ are provided in appendix F.3.

Posterior

Based on these local approximations, the approximate posterior can be written as

$$\begin{aligned}
\mathbb{P}(\mathbf{f}|\mathbf{y}, \mathbf{X}, \boldsymbol{\theta}) &\approx \mathcal{N}(\mathbf{f}|\mathbf{m}, \mathbf{V}) = \mathcal{N}\left(\mathbf{f}|\mathbf{m}, \left(\mathbf{K}^{-1} + \mathbf{W}\right)^{-1}\right) \\
\mathbf{W} &= -2\boldsymbol{\Lambda} \\
\mathbf{m} &= \mathbf{K}\boldsymbol{\alpha}
\end{aligned}$$

Marginal likelihood

Since the method inherently maximises a lower bound on the marginal likelihood, this bound (equation 4.13) is used as approximation to the marginal likelihood.

4.6 Individual potential bounding (VB)

Individual non-Gaussian likelihood bounds [Gibbs and MacKay, 2000, Jaakkola and Jordan, 1996] lead to many desirable properties for log-concave super-Gaussian models as described in chapter 3. The potential bounding approach can be seen as a variant of the KL method with more constraints or equivalently as a further relaxation to $\ln Z_{KL}$ (see 2.5.9). However, the convexity results only hold true for a special parametrisation in terms of the effective variance γ_i of the Gaussian approximation to the non-Gaussian likelihood $\mathbb{P}(y_i|f_i)$. We will first discuss a more general parametrisation and show how to deal with the cumulative Gaussian likelihoods. Then, we will add the respective expressions for the marginal likelihood using the analytically convenient special case.

Bounds

In general, every Gaussian lower bound has three variational parameters a_i, b_i and c_i

$$\begin{aligned}
\mathbb{P}(y_i|f_i) &\geq \exp\left(a_i f_i^2 + b_i y_i f_i + c_i\right), \; \forall f_i \in \mathbb{R}\, \forall i \qquad (4.19) \\
\Rightarrow \mathbb{P}(\mathbf{y}|\mathbf{f}) &\geq \exp\left(\mathbf{f}^\top\mathbf{A}\mathbf{f} + (\mathbf{b} \odot \mathbf{y})^\top \mathbf{f} + \mathbf{c}^\top\mathbf{1}\right) =: \mathbb{Q}(\mathbf{y}|\mathbf{f}, \mathbf{A}, \mathbf{b}, \mathbf{c}), \; \forall \mathbf{f} \in \mathbb{R},
\end{aligned}$$

where $\mathbf{A} = [a_i]_{ii}$, $\mathbf{b} = [b_i]_i$ and $\mathbf{c} = [c_i]_i$. It is clear, that a_i, b_i, c_i are not independent. Fixing one of them, more or less determines the others. Two possible parametrisations have been discussed in the literature: Gibbs and MacKay [2000] use $\varsigma \mapsto (\mathbf{A}, \mathbf{b}, \mathbf{c})$, where ς is the position at which the lower bound touches the likelihood and in Nickisch and Seeger [2009] we employ $\gamma \mapsto (\mathbf{A}, \mathbf{b}, \mathbf{c})$, where γ is the width of the lower bound. While the first parametrisation allows to deal with the cumulative Gaussian, the second parametrisation leads to a convex optimisation problem for the cumulative logistic likelihood. Table 4.1 summarises the parametrisation in terms of the positions ς and the widths γ.

Name	$\mathbf{A}$	$\mathbf{b}$	$\mathbf{c}$	tight at	notes
Cumulative logistic	$-\mathbf{\Lambda}_\varsigma$	$\frac{1}{2}\mathbf{1}$	$\mathbf{\Lambda}_\varsigma \varsigma^2 - \frac{1}{2}\varsigma + \ln \mathrm{sig}_{\mathrm{logit}}(\varsigma)$	$\mathbf{f} = \pm\varsigma$	$\lambda_i(\varsigma_i) = \frac{2\mathrm{sig}_{\mathrm{logit}}(\varsigma_i)-1}{4\varsigma_i}$
Cumulative Gaussian	$-\frac{1}{2}\mathbf{I}$	$\varsigma + \frac{\mathcal{N}(\varsigma)}{\mathrm{sig}_{\mathrm{probit}}(\varsigma)}$	$(\frac{\varsigma}{2} - \mathbf{b}) \odot \varsigma + \ln\left(\mathrm{sig}_{\mathrm{probit}}(\varsigma)\right)$	$\mathbf{f} = \varsigma$	see appendix F.8
Width based	$-\sigma^2 \frac{1}{2}\mathbf{\Gamma}^{-1}$	$\sigma^{-2}\boldsymbol{\beta} \odot \mathbf{y}$	$-\frac{1}{2}[h(\gamma_i)]_i$		$\sigma = 1$, see chapters 2/3

Table 4.1: Variational Bayes parametrisations

Posterior

Based on these two types of local bounds, the approximate posterior can be written as

$$\begin{aligned}
\mathbb{P}(\mathbf{f}|\mathbf{y}, \mathbf{X}, \boldsymbol{\theta}) &\approx \mathcal{N}(\mathbf{f}|\mathbf{m}, \mathbf{V}) = \mathcal{N}\left(\mathbf{f}|\mathbf{m}, \left(\mathbf{K}^{-1} + \mathbf{W}\right)^{-1}\right) \\
\mathbf{W} &= -2\mathbf{A}_\varsigma = \mathbf{\Gamma}^{-1} \\
\mathbf{m} &= \mathbf{V}(\mathbf{y} \odot \mathbf{b}_\varsigma) = \mathbf{V}\boldsymbol{\beta}_\gamma,
\end{aligned}$$

where we have expressed the posterior parameters directly as a function of the coefficients. Finally, we deal with an approximate posterior $\mathbb{Q}(\mathbf{f}|\boldsymbol{\theta}) = \mathcal{N}(\mathbf{f}|\mathbf{m}, \mathbf{V})$ only depending on a vector ς or γ of n variational parameters and a mapping $\varsigma, \gamma \mapsto (\mathbf{m}, \mathbf{V})$. In the KL method, every combination of values $\mathbf{m}$ and $\mathbf{W}$ is allowed, in the VB method, $\mathbf{m}$ and $\mathbf{V}$ cannot be chosen independently, since they have to be compatible with the bounding requirements. Therefore, the variational posterior is more constrained than the general Gaussian posterior.

Marginal likelihood

This lower bound on the individual likelihoods induces a lower bound on the marginal likelihood

$$Z = \int \mathbb{P}(\mathbf{f}|\mathbf{X}) \mathbb{P}(\mathbf{y}|\mathbf{f}) \, d\mathbf{f} \quad \geq \quad \int \mathbb{P}(\mathbf{f}|\mathbf{X}) \mathbb{Q}(\mathbf{y}|\mathbf{f}, \mathbf{A}, \mathbf{b}, \mathbf{c}) \, d\mathbf{f} = Z_{VB}.$$

Carrying out the Gaussian integral

$$Z_{VB} = \int \mathcal{N}(\mathbf{f}|0, \mathbf{K}) \exp\left(\mathbf{f}^\top \mathbf{A} \mathbf{f} + (\mathbf{b} \odot \mathbf{y})^\top \mathbf{f} + \mathbf{c}^\top \mathbf{1}\right) d\mathbf{f}$$

leads to (see appendix F.7)

$$\begin{aligned}
\ln Z_{VB} &= \mathbf{c}_\varsigma^\top \mathbf{1} + \frac{1}{2}(\mathbf{b}_\varsigma \odot \mathbf{y})^\top \left(\mathbf{K}^{-1} - 2\mathbf{A}_\varsigma\right)^{-1} (\mathbf{b}_\varsigma \odot \mathbf{y}) - \frac{1}{2}\ln|\mathbf{I} - 2\mathbf{A}_\varsigma \mathbf{K}| \qquad (4.20) \\
&= -\frac{1}{2}h(\gamma) + \frac{1}{2}\boldsymbol{\beta}^\top \left(\mathbf{K}^{-1} + \mathbf{\Gamma}^{-1}\right)^{-1} \boldsymbol{\beta} + \frac{1}{2}\ln|\mathbf{\Gamma}| - \frac{1}{2}\ln|\mathbf{K} + \mathbf{\Gamma}|,
\end{aligned}$$

which can now be maximised with respect to ς or γ. In order to get an efficient algorithm, we calculate the first and second derivatives $\partial \ln Z_{VB}/\partial\boldsymbol{\theta}$, $\partial \ln Z_{VB}/\partial\varsigma$, $\partial^2 \ln Z_{VB}/\partial\varsigma\partial\varsigma^\top$ (appendix F.1) and $\partial \ln Z_{VB}/\partial\gamma$, $\partial^2 \ln Z_{VB}/\partial\gamma\partial\gamma^\top$ (appendix F.2).

It turns out, that the approximation to the marginal likelihood (equation 4.20) is quite poor for the cumulative Gaussian likelihood and the more general Jensen bound approach (equation 4.13) is tighter.

4.7 Factorial variational method (FV)

Instead of approximating the posterior $\mathbb{P}(\mathbf{f}|\mathbf{y},\mathbf{X},\boldsymbol{\theta})$ by the closest Gaussian distribution, one can use the closest factorial distribution $\mathbb{Q}(\mathbf{f}|\mathbf{y},\mathbf{X},\boldsymbol{\theta}) = \prod_i \mathbb{Q}(f_i)$, also called *ensemble learning* [Csató et al., 2000, Miskin, 2000] . Another kind of factorial approximation $\mathbb{Q}(\mathbf{f}) = \mathbb{Q}(\mathbf{f}^+)\,\mathbb{Q}(\mathbf{f}^-)$ – a posterior factorising over classes – is used in multi-class classification [Girolami and Rogers, 2006].

Posterior

As a result of the free-form Kullback-Leibler divergence $\mathrm{KL}(\mathbb{Q}(\mathbf{f}|\mathbf{y},\mathbf{X},\boldsymbol{\theta}) \,\|\, \mathbb{P}(\mathbf{f}|\mathbf{y},\mathbf{X},\boldsymbol{\theta}))$ minimisation by equating its functional derivative $\delta\mathrm{KL}/\delta\mathbb{Q}(f_i)$ with the zero function (equation 2.14 and appendix F.9), one finds the best approximation to be of the following form

$$\begin{aligned} \mathbb{Q}(f_i) &\propto \mathcal{N}(f_i\,|\mu_i,\sigma_i^2)\,\mathbb{P}(y_i|f_i) \\ \mu_i &= m_i - \sigma_i^2\left[\mathbf{K}^{-1}\mathbf{m}\right]_i = [\mathbf{K}\boldsymbol{\alpha}]_i - \sigma_i^2\alpha_i \\ \sigma_i^2 &= \left[\mathbf{K}^{-1}\right]_{ii}^{-1} \\ m_i &= \int f_i\mathbb{Q}(f_i)\,\mathrm{d}f_i. \end{aligned} \tag{4.21}$$

In fact, the best product distribution consists of a factorial Gaussian times the original likelihood. The Gaussian has the same moments as the Leave-one-out prediction [Sundararajan and Keerthi, 2001]. Since the posterior is factorial, the effective likelihood of the factorial approximation has an odd shape. It effectively has to annihilate the correlations in the prior, and these correlations are usually what allows learning to happen in the first place. However, the best fitting factorial is still able to ensure that the latent means have the right signs. Even though all correlations are neglected, it is still possible that the model picks up the most important structure, since the expectations are coupled. Of course, at test time, it is essential that correlations are taken into account again using equation 4.10, as it would otherwise be impossible to inject any knowledge into the predictive distribution. For predictions we use the Gaussian $\mathcal{N}(\mathbf{f}|\mathbf{m},\mathrm{Dg}(\mathbf{v}))$ instead of $\mathbb{Q}(\mathbf{f})$. This is a further approximation, but it allows to stay inside the Gaussian framework.

Parameters μ_i and m_i are found by the following algorithm. Starting from $\mathbf{m} = \mathbf{0}$, iterate the following until convergence; (1) compute μ_i, (2) update m_i by taking a step in the direction towards m_i as given by equation 4.21. Step sizes are adapted.

Marginal likelihood

Surprisingly, one can obtain a lower bound on the marginal likelihood [Csató et al., 2000]

$$\ln Z \;\geq\; \sum_{i=1}^{n} \ln \mathrm{sig}\left(\frac{y_i m_i}{\sigma_i}\right) - \frac{1}{2}\boldsymbol{\alpha}^\top\left(\mathbf{K} - \mathrm{Dg}(\left[\sigma_1^2,\ldots,\sigma_n^2\right]^\top)\right)\boldsymbol{\alpha} - \frac{1}{2}\ln|\mathbf{K}| + \sum_{i=1}^{n}\ln\sigma_i.$$

4.8 Label regression method (LR)

Classification has also been treated using label regression or least squares classification [Rifkin and Klautau, 2004]. In its simplest form, this method simply ignores the discreteness of the class labels at the cost of not being able to provide proper probabilistic predictions. However, we treat LR as a heuristic way of choosing $\boldsymbol{\alpha}$ and $\mathbf{W}$, which allows us to think of it as yet another Gaussian approximation to the posterior allowing for valid predictions of class probabilities.

Posterior

After inference, according to equation 4.10, the moments of the (Gaussian approximation to the) posterior GP can be written as $\mu_* = \mathbf{k}_*^\top \boldsymbol{\alpha}$ and $\sigma_*^2 = k_{**} - \mathbf{k}_*^\top \left(\mathbf{K} + \mathbf{W}^{-1}\right)^{-1} \mathbf{k}_*$. Fixing

$$\mathbf{W}^{-1} = \sigma_n^2 \mathbf{I} \quad \text{and} \quad \boldsymbol{\alpha} = \left(\mathbf{K} + \mathbf{W}^{-1}\right)^{-1} \left(\mathbf{K} + \mathbf{W}^{-1}\right) \boldsymbol{\alpha} = \left(\mathbf{K} + \mathbf{W}^{-1}\right)^{-1} \mathbf{y},$$

we obtain GP regression from data points $\mathbf{x}_i \in \mathcal{X}$ to real labels $y_i \in \mathbb{R}$ with noise of variance σ_n^2 as a special case. In regression, the posterior moments are given by $\mu_* = \mathbf{k}_*^\top \left(\mathbf{K} + \sigma_n^2 \mathbf{I}\right)^{-1} \mathbf{y}$ and $\sigma_*^2 = k_{**} - \mathbf{k}_*^\top \left(\mathbf{K} + \sigma_n^2 \mathbf{I}\right)^{-1} \mathbf{k}_*$ [Rasmussen and Williams, 2006]. The arbitrary scale of the discrete $\mathbf{y}$ can be absorbed by the hyperparameters. There is an additional parameter σ_n, describing the width of the effective likelihood. In experiments, we selected $\sigma_n \in [0.5, 2]$ to maximise the log marginal likelihood.

Marginal likelihood

There are two ways of obtaining an estimate of the log marginal likelihood. One can simply ignore the binary nature and use the regression marginal likelihood $\ln Z_{\text{reg}}$ as proxy for $\ln Z$ – an approach we only mention but do not use in the experiments

$$\ln Z_{\text{reg}} = -\frac{1}{2} \boldsymbol{\alpha}^\top \left(\mathbf{K} + \sigma_n^2 \mathbf{I}\right) \boldsymbol{\alpha} - \frac{1}{2} \ln \left|\mathbf{K} + \sigma_n^2 \mathbf{I}\right| - \frac{n}{2} \ln 2\pi.$$

Alternatively, the Jensen bound (4.12) yields a lower bound $\ln Z \geq \ln Z_B$ – which seems more in line with the classification scenario than $\ln Z_{\text{reg}}$.

4.9 Relations between the methods

All considered approximations can be separated into *local* and *global methods*. Local methods exploit properties (such as derivatives) of the posterior at a special location only. Global methods minimise the KL-divergence $\text{KL}(\mathbb{Q}||\mathbb{P}) = \int \mathbb{Q}(\mathbf{f}) \ln \mathbb{Q}(\mathbf{f}) / \mathbb{P}(\mathbf{f})\, d\mathbf{f}$ between the posterior $\mathbb{P}(\mathbf{f})$ and a tractable family of distributions $\mathbb{Q}(\mathbf{f})$. Often this methodology is also referred to as a variational algorithm. Table 4.2 visualises the relations between the various algorithms.

assumption	relation	conditions	approx. posterior $\mathbb{Q}(\mathbf{f})$	name
$\mathbb{Q}(\mathbf{f}) = \mathcal{N}(\mathbf{f}\vert\mathbf{m}, \mathbf{V})$	$\rightarrow$	$\mathbf{m} = \arg\max_{\mathbf{f}} \mathbb{P}(\mathbf{f})$ $\mathbf{W} = -\frac{\partial^2 \ln \mathbb{P}(\mathbf{y}\vert\mathbf{f})}{\partial\mathbf{f}\partial\mathbf{f}^\top}$	$\mathcal{N}(\mathbf{f}\vert\mathbf{m}, (\mathbf{K}^{-1}+\mathbf{W})^{-1})$	LA
$\mathbb{Q}(\mathbf{f}) = \prod_i q_i(f_i)$	$\rightarrow$	$\frac{\delta \text{KL}}{\delta q_i(f_i)} \equiv \mathbf{0}$	$\prod_i \mathcal{N}(f_i\vert\mu_i, \sigma_i^2)\mathbb{P}(y_i\vert f_i)$	FV
	$\searrow$			
		$\left\langle f_i^d \right\rangle_{q_i(f_i)} = \left\langle f_i^d \right\rangle_{\mathbb{Q}(f_i)}$	$\mathcal{N}\left(\mathbf{f}\vert\mathbf{m}, (\mathbf{K}^{-1}+\mathbf{W})^{-1}\right)$	EP
	$\nearrow$			
$\mathbb{Q}(\mathbf{f}) = \mathcal{N}(\mathbf{f}\vert\mathbf{m}, \mathbf{V})$	$\rightarrow$	$\frac{\partial \text{KL}}{\partial \mathbf{V}, \mathbf{m}} = \mathbf{0}$	$\mathcal{N}\left(\mathbf{f}\vert\mathbf{m}, (\mathbf{K}^{-1}+\mathbf{W})^{-1}\right)$	KL
	$\searrow$			
$\mathbb{P}(y_i\vert f_i) \geq \mathcal{N}(f_i\vert\mu_{\varsigma_i}, \sigma_{\varsigma_i}^2)$	$\rightarrow$	$\frac{\partial \text{KL}}{\partial \varsigma_*} = \mathbf{0}$	$\mathcal{N}\left(\mathbf{f}\vert\mathbf{m}_{\varsigma_*}, (\mathbf{K}^{-1}+\mathbf{W}_{\varsigma_*})^{-1}\right)$	VB
$\mathbb{P}(y_i\vert f_i) := \mathcal{N}(f_i\vert y_i, \sigma_n^2)$	$\rightarrow$	$\mathbf{m} = (\mathbf{I}+\sigma_n^2\mathbf{K}^{-1})^{-1}\mathbf{y}$	$\mathcal{N}(\mathbf{f}\vert\mathbf{m}, (\mathbf{K}^{-1}+\sigma_n^{-2}\mathbf{I})^{-1})$	LR

Table 4.2: Relations between variational approximate inference algorithms

The only local method considered is the LA approximation matching curvature at the posterior mode. Common tractable distributions for global methods include factorial and Gaussian distributions. They have their direct correspondent in the FV method and the KL method. Individual likelihood bounds make the VB method a more constrained and easier-to-optimise version of the KL method. Interestingly, EP can be seen in some sense as a hybrid version of FV and KL, combining the advantages of both methods. Within the expectation consistence (EC) framework of Opper and Winther [2005], EP can be thought of as an algorithm that implicitly works with two distributions – a factorial and a Gaussian – having the same marginal moments $\langle f_i^d \rangle$. By means of iterative updates, one keeps these expectations consistent and produces a posterior approximation.

In the divergence measure and message passing framework of Minka [2005], EP is cast as a message passing algorithm template: iterative minimisation of local divergences to a tractable family of distributions yields a small global divergence. From that viewpoint, FV and KL are considered as special cases with divergence measure $\mathrm{KL}(\mathbb{Q}||\mathbb{P})$ combined with factorial and Gaussian distributions.

There is also a link between local and global methods, namely from the KL to the LA method. The necessary conditions for the LA method do hold *on average* for the KL method [Opper and Archambeau, 2009].

Finally, LR neither qualifies as local nor global – it is a heuristic way of setting **m** and **W**.

4.10 Markov chain Monte Carlo (MCMC)

The only way of getting a handle on the ground truth for the moments Z, $\mathbf{m}$ and $\mathbf{V}$ is by applying sampling techniques. In the limit of long runs, they are guaranteed to get the right answer. But in practise, these methods can be very slow, compared to analytic approximations discussed previously. MCMC runs are rather supposed to provide a gold standard for comparison with the other methods.

It turns out to be most challenging to obtain reliable marginal likelihood estimates as it is equivalent to solving the free energy problem in physics. We employ Annealed Importance Sampling (AIS) and thermodynamic integration to yield the desired marginal likelihoods. Instead of starting annealing from the prior distribution, we propose to directly start from an approximate posterior in order to speed up the sampling process.

Accurate estimates of the first and second moments can be obtained by sampling directly from the (unnormalised) posterior using Hybrid Monte Carlo methods [Neal, 1993].

Thermodynamic integration

The goal is to calculate the marginal likelihood $Z = \int \mathbb{P}(\mathbf{y}|\mathbf{f})\, \mathbb{P}(\mathbf{f}|\mathbf{X})\, \mathrm{d}\mathbf{f}$. AIS [Neal, 1993, 2001] works with intermediate quantities $Z_t := \int \mathbb{P}(\mathbf{y}|\mathbf{f})^{\tau(t)}\, \mathbb{P}(\mathbf{f}|\mathbf{X})\, \mathrm{d}\mathbf{f}$. Here, $\tau : \mathbb{N} \supset [0, T] \rightarrow [0, 1] \subset \mathbb{R}$ denotes an inverse temperature schedule with the properties $\tau(0) = 0$, $\tau(T) = 1$ and $\tau(t+1) \geq \tau(t)$ leading to $Z_0 = \int \mathbb{P}(\mathbf{f}|\mathbf{X})\, \mathrm{d}\mathbf{f} = 1$ and $Z_T = Z$.

On the other hand, we have $Z = Z_T/Z_0 = \prod_{t=1}^{T} Z_t/Z_{t-1}$ – an expanded fraction. Each factor Z_t/Z_{t-1} can be approximated by importance sampling with samples $\mathbf{f}_s$ from the "intermediate posterior" $\mathbb{P}(\mathbf{f}|\mathbf{y}, \mathbf{X}, t-1) := \mathbb{P}(\mathbf{y}|\mathbf{f})^{\tau(t-1)}\, \mathbb{P}(\mathbf{f}|\mathbf{X})\, / Z_{t-1}$ at time t.

$$\begin{aligned}
\frac{Z_t}{Z_{t-1}} &= \frac{\int \mathbb{P}(\mathbf{y}|\mathbf{f})^{\tau(t)}\, \mathbb{P}(\mathbf{f}|\mathbf{X})\, \mathrm{d}\mathbf{f}}{Z_{t-1}} = \int \frac{\mathbb{P}(\mathbf{y}|\mathbf{f})^{\tau(t)}}{\mathbb{P}(\mathbf{y}|\mathbf{f})^{\tau(t-1)}} \frac{\mathbb{P}(\mathbf{y}|\mathbf{f})^{\tau(t-1)}\, \mathbb{P}(\mathbf{f}|\mathbf{X})}{Z_{t-1}} \mathrm{d}\mathbf{f} \\
&= \int \mathbb{P}(\mathbf{y}|\mathbf{f})^{\Delta\tau(t)}\, \mathbb{P}(\mathbf{f}|\mathbf{y}, \mathbf{X}, t-1)\, \mathrm{d}\mathbf{f} \\
&\approx \frac{1}{S} \sum_{s=1}^{S} \mathbb{P}(\mathbf{y}|\mathbf{f}_s)^{\Delta\tau(t)}, \qquad \mathbf{f}_s \sim \mathbb{P}(\mathbf{f}|\mathbf{y}, \mathbf{X}, t-1) \qquad (4.22)
\end{aligned}$$

This works fine for small temperature changes $\Delta\tau(t) := \tau(t) - \tau(t-1)$. In the limit, we smoothly interpolate between $\mathbb{P}(\mathbf{y}|\mathbf{f})^0 \mathbb{P}(\mathbf{f}|\mathbf{X})$ and $\mathbb{P}(\mathbf{y}|\mathbf{f})^1 \mathbb{P}(\mathbf{f}|\mathbf{X})$, i.e. we start by sampling from the prior and finally approach the posterior. Note that sampling is algorithmically possible even though the distribution is only known up to a constant factor.

Improvement using an approximate posterior

In practise, the posterior can be quite different from the prior. That means individual fractions Z_t/Z_{t-1} may be difficult to estimate. One can make these fractions more similar by increasing the number of steps T or by "starting" from a distribution close to the posterior rather than from the prior. Let $\mathbb{Q}(\mathbf{f}) = \mathcal{N}(\mathbf{f}|\mathbf{m}, \mathbf{V}) \approx \mathbb{P}(\mathbf{f}|\mathbf{y}, \mathbf{X}, T) = \mathbb{P}(\mathbf{y}|\mathbf{f})\mathbb{P}(\mathbf{f}|\mathbf{X})/Z_T$ denote an approximation to the posterior. Setting $\mathcal{N}(\mathbf{f}|\mathbf{m}, \mathbf{V}) = \mathbb{Q}(\mathbf{y}|\mathbf{f})\mathbb{P}(\mathbf{f}|\mathbf{X})$, one can calculate the effective likelihood $\mathbb{Q}(\mathbf{y}|\mathbf{f})$ by division (see appendix F.5).

For the integration we use $Z_t = \int \mathbb{P}(\mathbf{y}|\mathbf{f})^{\tau(t)} \mathbb{Q}(\mathbf{y}|\mathbf{f})^{1-\tau(t)} \mathbb{P}(\mathbf{f}|\mathbf{X})\,\mathrm{d}\mathbf{f}$, where the Gaussian integral $Z_0 = \int \mathbb{Q}(\mathbf{y}|\mathbf{f})\mathbb{P}(\mathbf{f}|\mathbf{X})\,\mathrm{d}\mathbf{f}$ can be computed analytically. Again, each factor $\frac{Z_t}{Z_{t-1}}$ of the expanded fraction can be approximated by importance sampling from the modified intermediate posterior

$$\begin{aligned}\mathbb{P}(\mathbf{f}|\mathbf{y}, \mathbf{X}, t-1) &= \mathbb{P}(\mathbf{y}|\mathbf{f})^{\tau(t-1)} \mathbb{Q}(\mathbf{y}|\mathbf{f})^{1-\tau(t-1)} \mathbb{P}(\mathbf{f}|\mathbf{X})/Z_{t-1} \\ &= \left[\frac{\mathbb{P}(\mathbf{y}|\mathbf{f})}{\mathbb{Q}(\mathbf{y}|\mathbf{f})}\right]^{\tau(t-1)} \mathbb{Q}(\mathbf{y}|\mathbf{f})\mathbb{P}(\mathbf{f}|\mathbf{X})/Z_{t-1}.\end{aligned}$$

$$\begin{aligned}\frac{Z_t}{Z_{t-1}} &= \frac{\int \mathbb{P}(\mathbf{y}|\mathbf{f})^{\tau(t)} \mathbb{Q}(\mathbf{y}|\mathbf{f})^{1-\tau(t)} \mathbb{P}(\mathbf{f}|\mathbf{X})\,\mathrm{d}\mathbf{f}}{Z_{t-1}} \\ &= \int \frac{\mathbb{P}(\mathbf{y}|\mathbf{f})^{\tau(t)} \mathbb{Q}(\mathbf{y}|\mathbf{f})^{1-\tau(t)}}{\mathbb{P}(\mathbf{y}|\mathbf{f})^{\tau(t-1)} \mathbb{Q}(\mathbf{y}|\mathbf{f})^{1-\tau(t-1)}} \frac{\mathbb{P}(\mathbf{y}|\mathbf{f})^{\tau(t-1)} \mathbb{Q}(\mathbf{y}|\mathbf{f})^{1-\tau(t-1)} \mathbb{P}(\mathbf{f}|\mathbf{X})}{Z_{t-1}} \mathrm{d}\mathbf{f} \\ &= \int \left[\frac{\mathbb{P}(\mathbf{y}|\mathbf{f})}{\mathbb{Q}(\mathbf{y}|\mathbf{f})}\right]^{\Delta\tau(t)} \mathbb{P}(\mathbf{f}|\mathbf{y}, \mathbf{X}, t-1)\,\mathrm{d}\mathbf{f} \\ &\approx \frac{1}{S}\sum_{s=1}^{S} \left[\frac{\mathbb{P}(\mathbf{y}|\mathbf{f}_s)}{\mathbb{Q}(\mathbf{y}|\mathbf{f}_s)}\right]^{\Delta\tau(t)}, \qquad \mathbf{f}_s \sim \mathbb{P}(\mathbf{f}|\mathbf{y}, \mathbf{X}, t-1)\end{aligned}$$

The choice of $\mathbb{Q}(\mathbf{f})$ to be a good approximation to the true posterior makes the fraction $\mathbb{P}(\mathbf{y}|\mathbf{f})/\mathbb{Q}(\mathbf{y}|\mathbf{f})$ as constant as possible, which in turn reduces the error due to the finite step size in thermodynamical integration.

Algorithm

If only one sample $\mathbf{f}_t$ is used per temperature $\tau(t)$, the value of the entire fraction is obtained as

$$\ln \frac{Z_t}{Z_{t-1}} = \Delta\tau(t)\left[\ln \mathbb{P}(\mathbf{y}|\mathbf{f}_t) - \ln \mathbb{Q}(\mathbf{y}|\mathbf{f}_t)\right]$$

giving rise to the full estimate

$$\ln Z \approx \sum_{t=1}^{T} \ln \frac{Z_t}{Z_{t-1}} = \ln Z_{\mathbb{Q}} + \sum_{t=1}^{T} \Delta\tau(t)\left[\ln \mathbb{P}(\mathbf{y}|\mathbf{f}_t) + \frac{1}{2}(\mathbf{f}_t - \tilde{\mathbf{m}})^\top \mathbf{W}(\mathbf{f}_t - \tilde{\mathbf{m}})\right]$$

for a single run r. The finite temperature change bias can be removed by combining results Z_r from R different runs by their arithmetic mean $\frac{1}{R}\sum_r Z_r$ [Neal, 2001].

$$\ln Z = \ln \int \mathbb{P}(\mathbf{y}|\mathbf{f})\mathbb{P}(\mathbf{f}|\mathbf{X})\,\mathrm{d}\mathbf{f} \approx \ln\left(\frac{1}{R}\sum_{r=1}^{R} Z_r\right)$$

Finally, the only primitive needed to obtain MCMC estimates of Z, $\mathbf{m}$ and $\mathbf{V}$ is an efficient sampler for the "intermediate" posterior $\mathbb{P}(\mathbf{f}|\mathbf{y}, \mathbf{X}, t-1)$. We use Hybrid Monte Carlo sampling [Neal, 1993].

Results

If the posterior is very close to the prior (as in regimes 7-9 of figure 4.3), it does not make a difference, where we start from. However, if the posterior can be well approximated by a Gaussian (regimes 4-6), but is sufficiently different from the prior, then the method decreases variance and consequently improves runtimes of AIS. Different approximation methods lead also to differences in the improvement. Namely, the Laplace approximation performs worse than the approximation found by expectation propagation because Laplace's method approximates around the mode, which can be far away from the mean.

However, for our evaluations of the approximations to the marginal likelihood, we started the algorithm from the prior. Otherwise, one might be worried of biasing the MCMC simulation towards the initial distribution in cases of the chain failing to mix properly.

4.11 Implementation

As an extension to their book, Rasmussen and Williams [2006] made the GPML (Gaussian processes for machine learning) code publicly available[4]. The toolbox contained code for Gaussian regression and approximate classification using EP and LA. About a year later, the code was refactored and improved so that inference and model specification were kept apart[5]. In addition to EP and LA named `approxEP.m` and `approxLA.m` in the code, implementations of all of the approximation methods mentioned in this chapter can be downloaded and used[6] as an extension to the GPML code:

- `approxKL.m` – Kullback-Leibler, section 4.5,
- `approxVB.m` – individual variational bounds, section 4.6,
- `approxFV.m` – factorial variational, section 4.7 and
- `approxLR.m` – label regression 4.8.

Sparse and/or online approximation methods as introduced in section 4.2.2 include

- `approxIVM.m` – informative vector machine,
- `approxOLEP.m` – online EP and
- `approxSO.m` – sparse online approximation.

For mainly educational reasons, we also provide some (equivalent) variants of EP from section 4.4.1 like

- `approxEC.m` – expectation consistent inference,
- `approxTAP.m` – ADATAP and
- `approxTAPnaive.m` – naive ADATAP.

The release 3.1 of the GPML code as described in section 4.11.1 [Rasmussen and Nickisch, 2010], is available as `mloss.org` project[7] or from the Gaussian process website[8]. The new implementation is completely generic, with simple interfaces for an extended set of covariance and likelihood functions. We also support arbitrary mean functions and provide full compatibilty with GNU Octave. Much energy was spent to properly disentangle covariance, likelihood and mean hyperparameters. Again, special care has been taken to avoid numerical problems, e.g. safe likelihood evaluations for extreme inputs and stable matrix operations as described in the following.

[4] `http://www.gaussianprocess.org/gpml/code/matlab/release/gpml-matlab-v1.3-2006-09-08.tar.gz`

[5] `http://www.gaussianprocess.org/gpml/code/matlab/release/gpml-matlab-v2.0-2007-06-25.tar.gz`

[6] The extension is available at `http://www.kyb.mpg.de/~hn/approxXX.tar.gz`.

[7] `http://mloss.org/software/view/263/`

[8] The current version can be obtained from `http://www.gaussianprocess.org/gpml/code/matlab/doc/`.

Stable matrix operations

More concretely, to properly handle situations, where $\mathbf{K}$ is close to singular, we use the well-conditioned matrix $\mathbf{B}$[9] and its Cholesky decomposition to calculate $\mathbf{V} = \left(\mathbf{K}^{-1} + \mathbf{W}\right)^{-1}$ and $\mathbf{k}_*^\top \mathbf{C}\mathbf{k}_* = \mathbf{k}_*^\top \left(\mathbf{K} + \mathbf{W}^{-1}\right)^{-1} \mathbf{k}_*$. The case of $\mathbf{W}$[10] having negative components, can be handled by using the (slower) LU-decomposition of the non-symmetric (but well-conditioned) matrix $\mathbf{A}$ instead as summarised in the following table.

Well conditioned matrix	$\mathbf{C} = \left(\mathbf{K} + \mathbf{W}^{-1}\right)^{-1}$	$\mathbf{V} = \left(\mathbf{K}^{-1} + \mathbf{W}\right)^{-1}$	$\ln \lvert\mathbf{KW} + \mathbf{I}\rvert$
$\mathbf{B} = \mathbf{W}^{\frac{1}{2}}\mathbf{K}\mathbf{W}^{\frac{1}{2}} + \mathbf{I} = \mathbf{L}\mathbf{L}^\top$	$\mathbf{W}^{\frac{1}{2}}\mathbf{B}^{-1}\mathbf{W}^{\frac{1}{2}} = \mathbf{W}^{\frac{1}{2}}\mathbf{L}^{-\top}\mathbf{L}^{-1}\mathbf{W}^{\frac{1}{2}}$	$\mathbf{K} - \mathbf{KCK}$	$2 \cdot \mathbf{1}^\top \ln(\mathrm{dg}(\mathbf{L}))$
$\mathbf{A} = \mathbf{KW} + \mathbf{I} = \mathbf{LU}$	$\mathbf{W}\mathbf{A}^{-1} = \mathbf{W}\mathbf{U}^{-1}\mathbf{L}^{-1}$	$\mathbf{K} - \mathbf{KCK}$	$\mathbf{1}^\top \ln \lvert\mathrm{dg}(\mathbf{U})\rvert$

Table 4.3: Numerically stable matrix operations in GP classification

The posterior mean $\mathbf{m}$ is represented in terms of $\boldsymbol{\alpha} = \mathbf{K}^{-1}\mathbf{m}$ to avoid multiplications with $\mathbf{K}^{-1}$ and facilitate predictions. As a result, our code shows a high level of robustness along the full spectrum of possible hyperparameters. The KL method uses Gaussian-Hermite quadrature; we did not notice problems stemming therefrom. The FV and TAP methods work very reliably, although we had to add a small (10^{-6}) ridge for FV to regularise $\mathbf{K}$.

Large scale computations

The focus of the toolbox is on approximate inference using dense matrix algebra. We currently do not support covariance matrix approximation techniques to deal with large numbers of training examples n. Hence, all discussed inference algorithms hinge on $\mathbf{K}$ being not too big since matrix decompositions have complexity $\mathcal{O}(n^3)$. If the dataset size n grows beyond $5 \cdot 10^3$, exact matrix computations become prohibitive rather quickly. By means of an approximation to the covariance matrix

$$\mathbf{K} \approx \hat{\mathbf{K}} := \mathbf{V}\mathbf{R}\mathbf{V}^\top + \mathbf{D},\ \mathbf{V} \in \mathbb{R}^{n\times r},\ \mathbf{R} \in \mathbb{R}^{r\times r},\ \mathbf{D} = \mathrm{dg}(\mathbf{d}),$$

which has to be computed before the inference procedure, we can reduce the computational cost so that LA and VB become scalable. Examples include the Nyström approximation [Smola and Schölkopf, 2000, Williams and Seeger, 2001] and the incomplete Cholesky decomposition [Fine and Scheinberg, 2001]. Matrix vector multiplications (MVMs) with $\hat{\mathbf{K}}$ cost $\mathcal{O}(r \cdot n)$ instead of $\mathcal{O}(n^2)$ and MVMs with $\hat{\mathbf{K}}^{-1}$ can be computed using the matrix inversion lemma

$$\mathbf{K}^{-1} \approx \hat{\mathbf{K}}^{-1} = \mathbf{D}^{-1} - \mathbf{D}^{-1}\mathbf{V}\left(\mathbf{R}^{-1} - \mathbf{V}^\top\mathbf{D}^{-1}\mathbf{V}\right)^{-1}\mathbf{V}^\top\mathbf{D}^{-1}$$

at a cost of $\mathcal{O}(r \cdot n)$, as well.

4.11.1 The `gpml` toolbox

We provide a stable and modular implementation verified by test cases and unit tests that contains a user and a technical documentation[11]. The code is fully compatible to Matlab 7.x[12] and GNU Octave 3.2.x[13]. A Gaussian process model requires the specification of a Gaussian process prior through a mean and covariance function and as well as a likelihood. Model fitting and prediction depends on an approximate inference algorithm computing $\mathbb{Q}(\mathbf{f})$ and $\mathbb{Q}(f_*)$ as summarised in the following table.

The GPML toolboox contains exactly these objects: model fitting using the marginal likelihood gradient $\frac{\partial L}{\partial \boldsymbol{\theta}}$ and prediction work in a fully generic way, once the model is specified. In the following, we list some of the implemented objects.

[9] All eigenvalues λ of $\mathbf{B}$ satisfy $1 \leq \lambda \leq 1 + \frac{n}{4}\max_{ij}\mathbf{K}_{ij}$, thus $\mathbf{B}^{-1}$ and $\lvert\mathbf{B}\rvert$ can be safely computed.

[10] This happens for non-log-concave likelihoods like the Student's t likelihood. Formally, negative values in $\mathbf{W}$ correspond to negative variances. Although negative variances do not have a probabilistic meaning, they still allow to locally imitate the non-Gaussian likelihoods so that the approximate posterior is most similar to the exact

1) GP $f \sim \mathcal{GP}\left(m_{\boldsymbol{\phi}}, k_{\boldsymbol{\psi}}\right)$	2) Likelihood	3) Approximate inference	4) Fitting $\boldsymbol{\theta} = (\boldsymbol{\phi}, \boldsymbol{\psi}, \boldsymbol{\rho})$
$\mathbb{P}_{\boldsymbol{\phi},\boldsymbol{\psi}}(\mathbf{f}) \sim \mathcal{N}(\mathbf{f}\|\mathbf{m}_{\boldsymbol{\phi}}, \mathbf{K}_{\boldsymbol{\psi}})$	$\prod_{i=1}^{n} \mathbb{P}_{\rho}(y_i\|f_i)$	$\mathbb{Q}(\mathbf{f}) \approx \mathbb{P}(\mathbf{f}\|\mathbf{y}) \propto \mathbb{P}_{\boldsymbol{\phi},\boldsymbol{\psi}}(\mathbf{f})\mathbb{P}_{\boldsymbol{\rho}}(\mathbf{y}\|\mathbf{f})$	$\boldsymbol{\theta}^{\star} = \arg\max_{\boldsymbol{\theta}} L(\boldsymbol{\theta})$
a) mean $m_{\boldsymbol{\phi}}(\mathbf{x})$		$L(\boldsymbol{\theta}) \approx \int \mathbb{P}_{\boldsymbol{\phi},\boldsymbol{\psi}}(\mathbf{f})\mathbb{P}_{\boldsymbol{\rho}}(\mathbf{y}\|\mathbf{f})\mathrm{d}\mathbf{f}$	5) Prediction $\mathbb{Q}(y_*)$
b) covariance $k_{\boldsymbol{\psi}}(\mathbf{x}, \mathbf{x}')$			

Table 4.4: GPML toolbox building blocks

1a) Mean functions

In the GPML toolbox a mean function needs to implement evaluation $\mathbf{m} = m_{\boldsymbol{\phi}}(\mathbf{X})$ and first derivatives $\mathbf{m}_i = \frac{\partial}{\partial \phi_i} m_{\boldsymbol{\phi}}(\mathbf{X})$. We offer simple and composite mean functions.

- simple functions: `zero` $m(\mathbf{x}) = 0$, `const` $m(\mathbf{x}) = c$, `linear` $m(\mathbf{x}) = \mathbf{a}^\top \mathbf{x}$
- composite functions: `sum` $m(\mathbf{x}) = \sum_j m_j(\mathbf{x})$, `prod` $m(\mathbf{x}) = \prod_j m_j(\mathbf{x})$, `pow` $m(\mathbf{x}) = m_1(\mathbf{x})^d$

This modular specification allows to work with affine mean functions $m(\mathbf{x}) = c + \mathbf{a}^\top \mathbf{x}$ or polynomials $m(\mathbf{x}) = (c + \mathbf{a}^\top \mathbf{x})^2$.

1b) Covariance functions

Similarly to the mean functions, we provide a whole algebra of covariance functions. Again, the interface is simple since only evaluation of the full covariance matrix $\mathbf{K} = k_{\boldsymbol{\psi}}(\mathbf{X})$ and its derivatives $\mathbf{K}_i = \frac{\partial}{\partial \psi_i} k_{\boldsymbol{\psi}}(\mathbf{X})$ and cross terms $\mathbf{k}_* = k_{\boldsymbol{\psi}}(\mathbf{X}, \mathbf{x}_*)$ and $k_{**} = k_{\boldsymbol{\psi}}(\mathbf{x}_*, \mathbf{x}_*)$ for prediction are required. Besides a long list of simple covariance functions, we also offer a variety of composite covariance functions.

- simple functions: `linear, constant, ridge, Matérn, squared exponential, polynomial, periodic, MKL, neural network, finite support`
- composite functions
  - `sum, prod` $k(\mathbf{x}, \mathbf{x}') = \sum_j k_j(\mathbf{x}, \mathbf{x}'), k(\mathbf{x}, \mathbf{x}') = \prod_j k_j(\mathbf{x}, \mathbf{x}')$
  - `masked` $k(\mathbf{x}_I, \mathbf{x}'_I)$, masking index $I \subseteq [1, 2, .., D]$, $\mathbf{x} \in \mathbb{R}^D$
  - `scaling` $k(\mathbf{x}, \mathbf{x}) = \sigma_f^2 k_0(\mathbf{x}, \mathbf{x}')$
  - `additive` $k(\mathbf{x}, \mathbf{x}') = \sum_{|I|=d \in \mathcal{D}} k(\mathbf{x}_I, \mathbf{x}'_I)$, index degree set $\mathcal{D}$

Both the mean and the covariance functions are easily extensible.

2) Likelihoods

The GPML toolbox approximate inference engine does not explicitly distinguish between classification and regression: for any choice of the likelihood $\mathbb{P}_\rho(y_i|f_i)$, the toolbox uses the same code in the inference step. The following table enumerates all (currently) implemented likelihood functions and their respective parameter set $\boldsymbol{\rho}$. See figure 2.2 for a graphical illustration and the expressions for $\mathbb{P}_\rho(y_i|f_i)$.

posterior.

[11] `http://www.gaussianprocess.org/gpml/code/`

[12] The MathWorks, `http://www.mathworks.com/`

[13] The Free Software Foundation, `http://www.gnu.org/software/octave/`

$\mathbb{P}_\rho(y_i\|f_i)$	regression $y_i \in \mathbb{R}$	classification $y_i \in \{\pm 1\}$				
name	Gaussian	logistic	Laplacian	Student's t	cum. Gaussian	cum. logistic
$\rho =$	$\{\ln \sigma\}$			$\{\ln(\nu - 1), \ln \sigma\}$	$\emptyset$	

Table 4.5: Likelihood functions implemented in the GPML toolbox

3) Approximate inference methods

In addition to exact inference (only possible for Gaussian likelihood), we have three major approximate inference methods implemented in the toolbox: expectation propagation (section 4.4 and chapter 2.5.10), Laplace approximation (section 4.3 and chapter 2.5.6) and variational Bayes (section 4.6 and chapter 2.5.9). The following table lists all possible combinations of likelihood and inference algorithm. Note that any choice of mean and covariance function is allowed.

likelihood \ inference	exact	EP	Laplace	variational Bayes
Gaussian	✓	✓	✓	
logistic		✓	✓	✓
Laplacian		✓		✓
Student's t			✓	✓
cumulative Gaussian		✓	✓	
cumulative logistic		✓	✓	✓

Table 4.6: Likelihood ↔ inference compatibility in the GPML toolbox

Expectation propagation for Student's t likelihoods is inherently unstable due to non-log-concavity. The Laplace approximation for Laplace likelihoods is not sensible because at the mode the curvature and the gradient can be undefined due to the non-differentiable peak of the Laplace distribution. Special care has been taken for the non-convex optimisation problem imposed by the combination Student's t likelihood and Laplace approximation. Finally, the (convex) lower bounding approach by Gaussian potentials of variable width is problematic for Gaussian and cumulative Gaussian likelihoods because they admit only certain widths.

Code example

Due to the modular structure of the code, specification of a full GP model and model fitting can be done in less than ten lines of code as illustrated by the following example.

```
[xtr,xte,ytr,yte] = read_data;                       % train and test data

% 1) SET UP THE GP
cov  = {'covSEiso'}; sf = 1; ell = 0.7;   % squared exp. covariance
mean = {'meanSum',{'meanLinear','meanConst'}}; a = 2; b = 1;% a*x+b
lik  = 'likLaplace'; sn = 0.2;           % sparse Laplace likelihood
hyp.mean = [a;b]; hyp.cov = log([ell;sf]); hyp.lik  = log(sn);% hyp
inf = 'infEP';          % inference method is expectation propagation

% 2) LEARN, i.e. MAX. MARGINAL LIKELIHOOD w.r.t. hyp
Ncg = 50;      % number of conjugate gradient steps for optimisation
hyp = minimize(hyp,'gp', -Ncg, inf, mean, cov, lik, xtr, ytr);

% 3) PREDICT
[ymu, ys2] = gp(hyp, inf, mean, cov, lik, xtr, ytr, xte)

```

```
K  = feval(cov{:},  hyp.cov, xtr);      % evaluate covariance matrix
m  = feval(mean{:}, hyp.mean, xtr);          % evaluate mean vector
lp = feval(lik, hyp.lik, ytr, ftr);        % evaluate log likelihood
```

4.12 Experiments

The purpose of our experiments is to illustrate the strengths and weaknesses of the different approximation methods. First of all, the quality of the approximation itself in terms of posterior moments Z, $\mathbf{m}$ and $\mathbf{V}$ is studied. At a second level, building on the "low-level" features, we compare predictive performance in terms of the predictive probability p_* given by (equations 4.8 and 4.10)

$$p_* := \mathbb{P}\left(y_* = 1|\mathbf{x}_*, \mathbf{y}, \mathbf{X}, \boldsymbol{\theta}\right) \quad \approx \quad \int \mathrm{sig}\left(f_*\right) \mathcal{N}\left(f_*|\mu_*, \sigma_*^2\right) \mathrm{d}f_*. \tag{4.23}$$

On a third level, we assess higher order properties such as the information score, describing how much information the model managed to extract about the target labels, and the error rate – a binary measure of whether a test input is assigned the right class. Uncertainty predictions provided by the model are not captured by the error rate.

Accurate marginal likelihood estimates Z are a key to hyperparameter learning. In that respect, Z can be seen as a high-level feature and as the "zeroth" posterior moment at the same time.

A summary of the results is provided by table 4.7.

Datasets

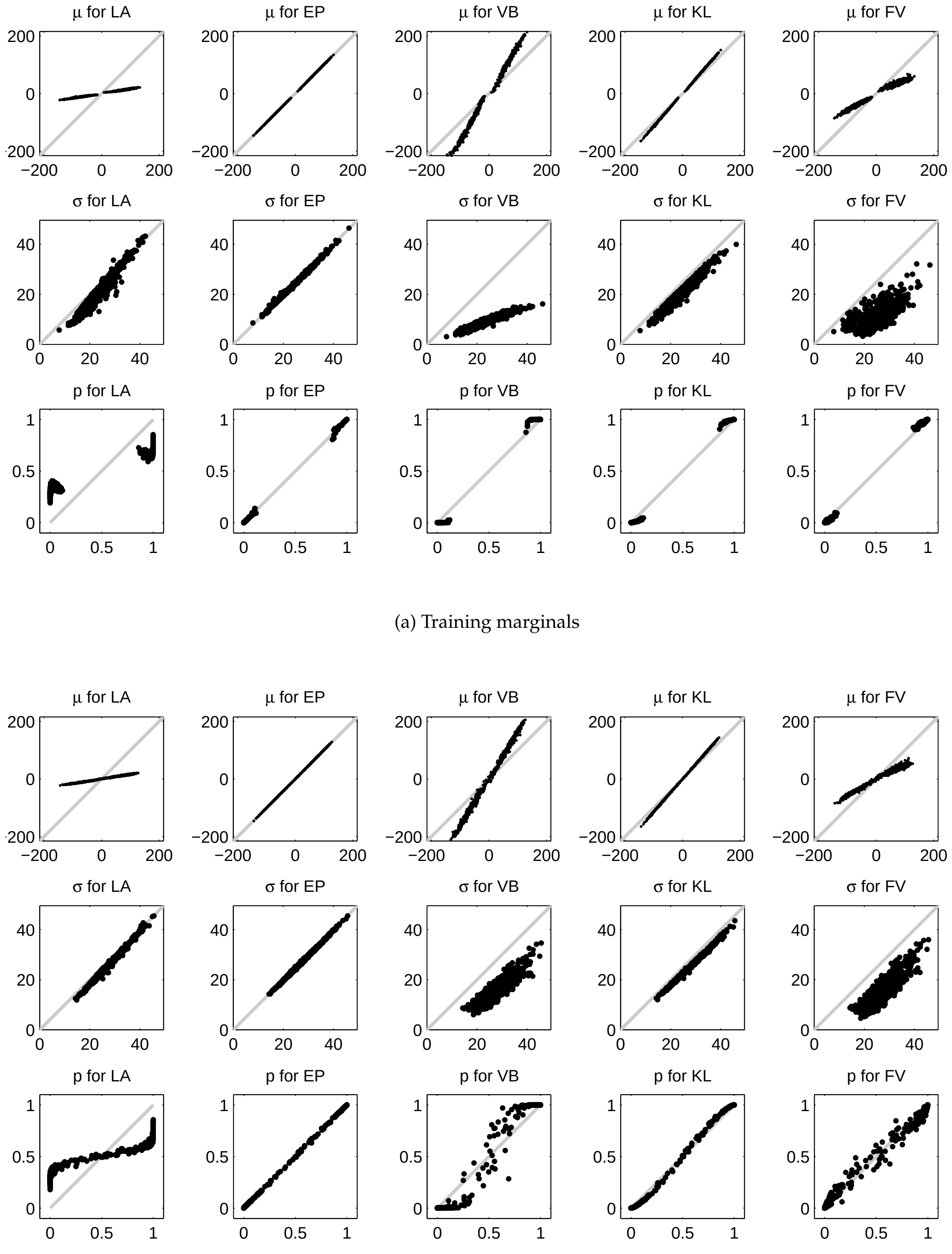
One main goal is to study the general behaviour of approximate GP classification. Our results for the different approximation methods are not specific to a particular dataset but apply to a wide range of application domains. This is reflected by the choice of our reference datasets summarised in table 4.8, widely used in the machine learning literature. We do not include the full experiments on all datasets. However, we have verified that the same qualitative conclusions hold for all the datasets considered.

Results

In the following, we report our experimental results covering posterior moments and predictive performance. Findings for all 5 methods are provided to make the methods as comparable as possible.

4.12.0.1 Mean m and (co)variance V

The posterior process, or equivalently the posterior distribution over the latent values $\mathbf{f}$, is determined by its location parameter $\mathbf{m}$ and its width parameter $\mathbf{V}$. In that respect, these two low-level quantities are the basis for all further calculations. In general, one can say that the methods show significant differences in the case of highly non-Gaussian posteriors (regimes 1-5 of figure 4.3). Even in the two-dimensional toy example of figures 4.4 and 4.5, significant differences are apparent. The means are inaccurate for LA and VB; whereas the variances are somewhat underestimated by LA and KL and severely so by VB. Marginal means $\mathbf{m}$ and variances $\mathrm{dg}(\mathbf{V})$ for USPS 3 vs. 5 are shown in figure 4.6; an exemplary marginal is pictured in figure 4.7 for all approximate methods and the MCMC estimate. Along the same lines, a close-to-Gaussian posterior is illustrated in figure 4.8. We chose the hyperparameters for the non-Gaussian case of figure 4.6 to maximise the EP marginal likelihood (see figure 4.9), whereas the hyperparameters of figure 4.8 were selected to yield a posterior that is almost Gaussian but still has reasonable predictive performance.



(a) Training marginals

(b) Test marginals

Figure 4.6: Marginals of USPS 3 vs. 5 for a highly non-Gaussian posterior
Each row consists of five plots showing MCMC ground truth on the x-axis and LA, EP, VB, KL and FV on the y-axis. Based on the cumulative logistic likelihood function and the squared exponential covariance function with parameters $\ln \ell = 2.25$ *and* $\ln \sigma_f = 4.25$ *we plot the marginal means, standard deviations and resulting predictive probabilities in rows 1-3. We are working in regime 2 of figure 4.3 that means the posterior is highly non-Gaussian. The upper part shows marginals of training points and the lower part shows test point marginals.*

	LA	EP*	VB logit | probit	KL	FV	MCMC
idea	quadratic expansion around the mode	marginal moment matching	lower bound on indiv. likelihoods	KL minim., average w.r.t. wrong $\mathbb{Q}(\mathbf{f})$	best free-form factorial	sampling, thermo-dynamic integration
algorithm	Newton steps	iterative matching	Newton steps	Newton steps	fixed-point iteration	Hybrid MC, AIS
complexity	$\mathcal{O}(n^3)$	$\mathcal{O}(n^3)$	$\mathcal{O}(n^3)$	$\mathcal{O}(8n^3)$	$\mathcal{O}(n^3)$	$\mathcal{O}(n^3)$
speed	very fast	fast	fast	slow	very fast	very slow
running time	1	10	8	150	4	>500
likelihood properties	1st-3rd log. derivative	$\mathcal{N}$-integrals	lower bound	simple evaluation	$\mathcal{N}$-integrals	1st log derivative
evidence Z	–	≈	– –	–	– – –	=
mean $\mathbf{m}$	– –	≈	++| – –	+	–	=
covariance $\mathbf{V}$	–	≈	– –	–	– –	=
information I	–	≈	≈| –	≈	–	=
PRO	speed	practical accuracy		principled method	speed	theoretical accuracy
CON	mean≠mode, low info I	speed	strong over-confidence	overconfidence	factorising approxima-tion	very slow

Table 4.7: Feature summary of the considered algorithms
For each of the six algorithms under consideration, the major properties are listed. The basic idea of the method along with its computational algorithm and complexity is summarised, the requirements to the likelihood functions are given, the accuracy of evidence and moment estimates as well as information is outlined and some striking advantages and drawbacks are compared. Six relations characterise accuracy: – – – extreme underestimation, – – heavy underestimation, – underestimation, = ground truth, ≈ good approximation, + overestimation and ++ heavy overestimation.
Running times were calculated by running each algorithm for 9 different hyperparameter regimes and both likelihoods on all datasets. An average running time per dataset was calculated for each method and scaled to yield 1 for LA. In the table, the average of these numbers is shown. We are well aware of the fact, that these numbers also depend on our Matlab implementations and choices of convergence thresholds.

The LA method has the principled weakness of expanding around the mode. In high-dimensional spaces, the mode can be very far away from the mean [Kuss and Rasmussen, 2005]. The absolute value of the mean is strongly underestimated. Furthermore, the posterior is highly curved at its mode, which leads to an underestimated variance, too. These effects can be seen in the first column of figures 4.6 and 4.7, although in the close-to-Gaussian regime LA works well, figure 4.8. For large latent function scales σ_f^2, in the limit $\sigma_f^2 \to \infty$, the likelihood becomes a step function, the mode approaches the origin and the curvature at the mode becomes larger. Thus the approximate posterior as found by LA becomes a zero-mean Gaussian which is much too narrow.

The EP method almost perfectly agrees with the MCMC estimates, second column of figure 4.6. That means, iterative matching of approximate marginal moments leads to accurate marginal moments of the posterior.

The KL method minimises the KL-divergence $\mathrm{KL}\left(\mathbb{Q}(\mathbf{f}) \,\|\, \mathbb{P}(\mathbf{f})\right) = \int \mathbb{Q}(\mathbf{f}) \ln \frac{\mathbb{Q}(\mathbf{f})}{\mathbb{P}(\mathbf{f})} \mathrm{d}\mathbf{f}$ with the average taken to the approximate distribution $\mathbb{Q}(\mathbf{f})$. The method is *zero-forcing*, i.e. in regions where $\mathbb{P}(\mathbf{f})$ is very small, $\mathbb{Q}(\mathbf{f})$ has to be very small as well. In the limit that means $\mathbb{P}(\mathbf{f}) = 0 \Rightarrow \mathbb{Q}(\mathbf{f}) = 0$. Thus, the support of $\mathbb{Q}(\mathbf{f})$ is smaller than the support of $\mathbb{P}(\mathbf{f})$ and

Dataset	n_{train}	n_{test}	d	Brief description of problem domain
Breast	300	383	9	Breast cancer[14]
Crabs	100	100	6	Sex of Leptograpsus crabs[15]
Ionosphere	200	151	34	Classification of radar returns from the ionosphere[16]
Pima	350	418	8	Diabetes in Pima Indians[17]
Sonar	108	100	60	Sonar signals bounced by a metal or rock cylinder[18]
USPS 3 vs. 5	767	773	256	Binary sub-problem of the USPS handwritten digit dataset[19]

Table 4.8: Dimensionality of the considered datasets

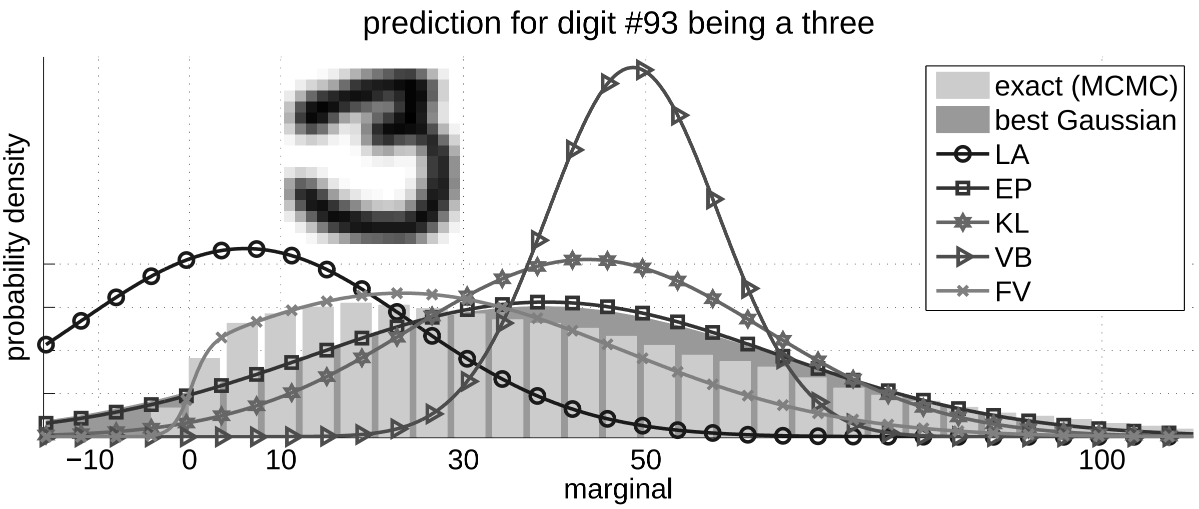


Figure 4.7: Marginals of USPS 3 vs. 5 for digit #93
Posterior marginals for one special training point from figure 4.6 is shown. Ground truth in terms of true marginal and best Gaussian marginal (matching the moments of the true marginal) are plotted in grey, Gaussian approximations are visualised as lines. For multivariate Gaussians $\mathcal{N}(\mathbf{m}, \mathbf{V})$, the i-th marginal is given by $\mathcal{N}([\mathbf{m}]_i, [\mathbf{V}]_{ii})$. Thus, the mode m_i of marginal i coincides with the i-th coordinate of the mode of the joint $[\mathbf{m}]_i$. This relation does not hold for general skewed distribution. Therefore, the marginal given by the Laplace approximation is not centred at the mode of the true marginal.

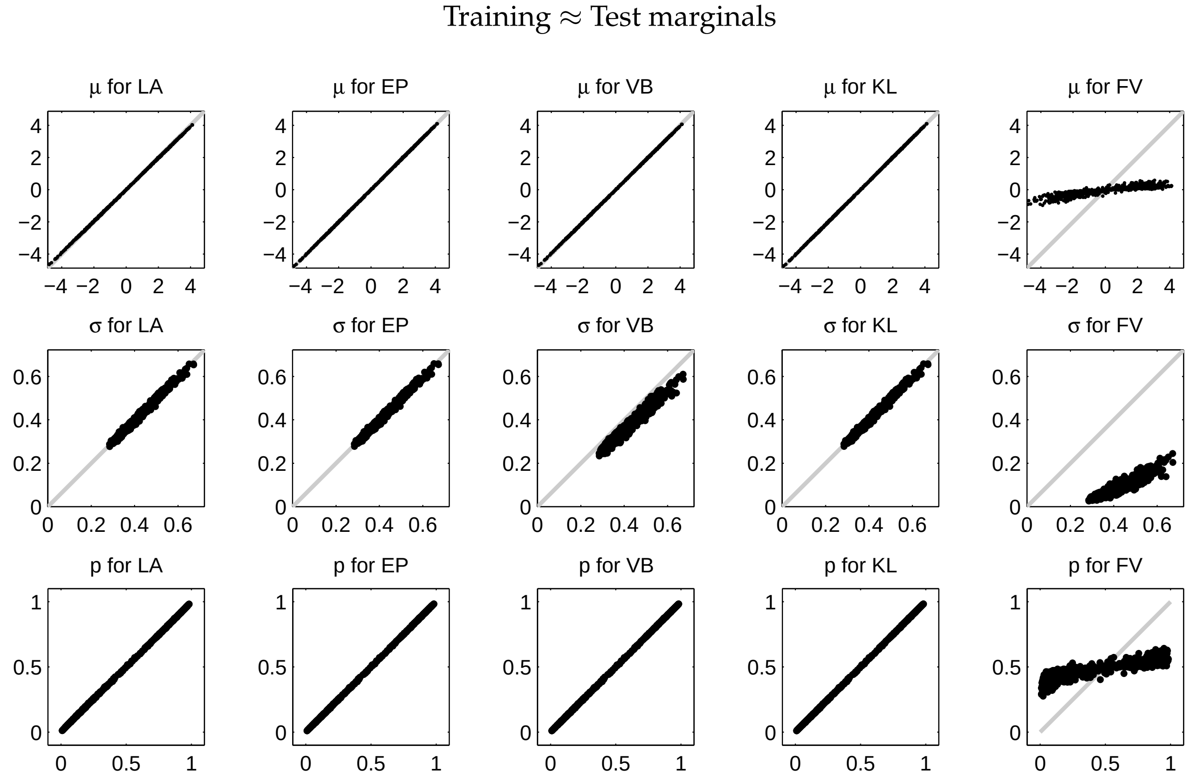
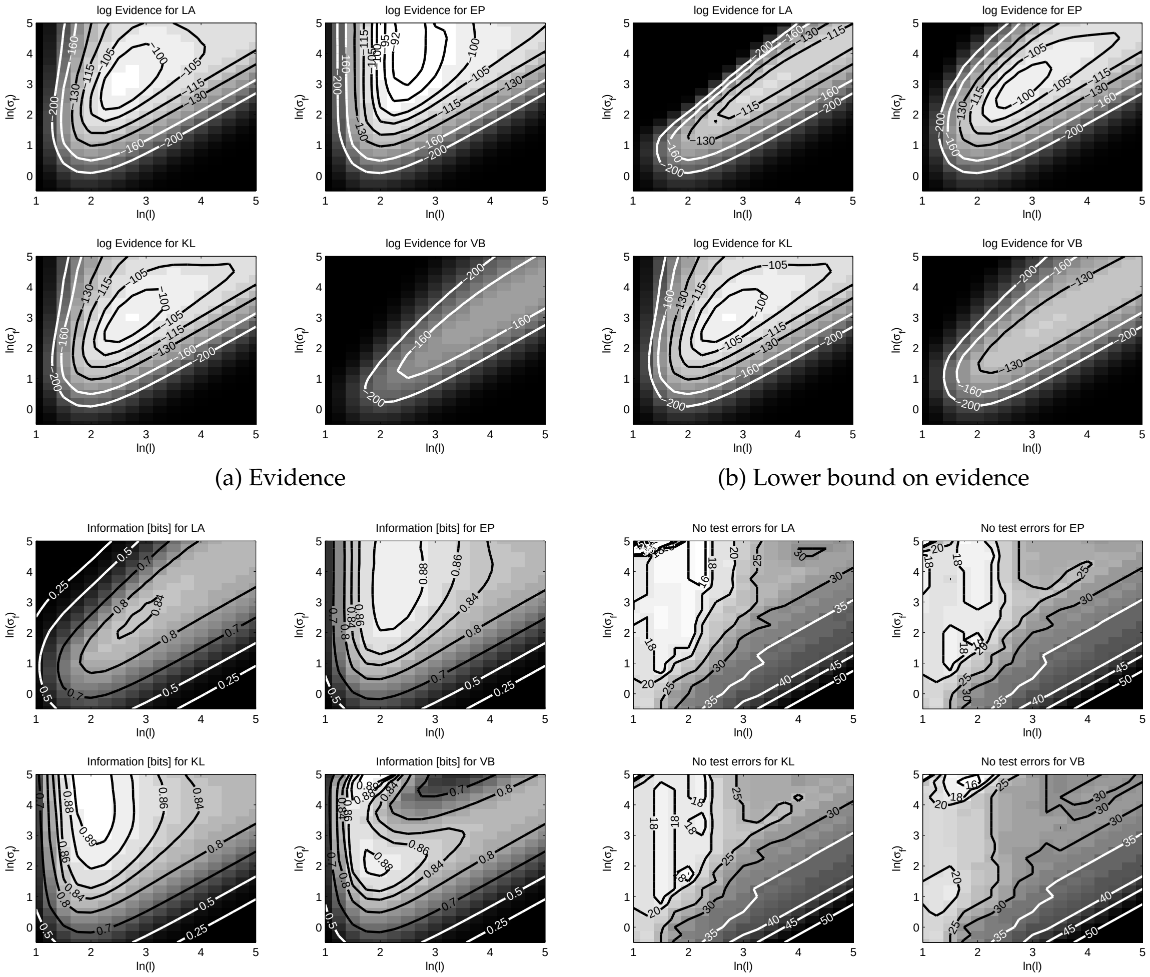


Figure 4.8: Marginals of USPS 3 vs. 5 for a close-to-Gaussian posterior
Using the squared exponential covariance and the cumulative logistic likelihood function with parameters $\ln \ell = 3$ *and* $\ln \sigma_f = 0.5$*, we plot the marginal means, standard deviations and resulting predictive probabilities in rows 1-3. Only the quantities for the trainingset are shown, because the test set results are very similar. We are working in regime 8 of figure 4.3 that means the posterior is of rather Gaussian shape. Each row consists of five plots showing MCMC ground truth on the x-axis and LA, EP, VB, KL and FV on the y-axis.*

hence the variance is underestimated. Typically, the posterior has a long tail away from zero as seen in figure 4.3 regimes 1-5. The zero forcing property shifts the mean of the approximation away from the origin, which results in a slightly overestimated mean, fourth column of figure 4.6.

Finally, the VB method can be seen as a more constrained version of the KL method with deteriorated approximation properties. The variance underestimation and mean overestimation is magnified, third column of figure 4.6. Due to the required lower bounding property of each individual likelihood term, the approximate posterior has to obey severe restrictions. Especially, the lower bound to the cumulative Gaussian cannot adjust its width since the asymptotic behaviour does not depend on the variational parameter (equation 4.19).

The FV method has a special role because it does not lead to a Gaussian approximation to the posterior but to the closest (in terms of KL-divergence) factorial distribution. If the prior is quite isotropic (regimes 1, 4 and 7 of figure 4.3), the factorial approximation provides a reasonable approximation. If the latent function values are correlated, the approximation fails. Because of the zero forcing property, mentioned in the discussion of the KL method, both the means and the variances are underestimated. Since a factorial distribution cannot capture correlations, the effect can be severe. It is worth mentioning that there is no difference whether the posterior is close to a Gaussian or not. In that respect, the FV method complements the LA method, which has difficulties in regimes 1, 2 and 4 of figure 4.3.



(a) Evidence

(b) Lower bound on evidence

(c) Information in bits

(d) Number of errors

Figure 4.9: Evidence and classification performance for LA, EP, KL & VB on USPS 3 vs. 5 The length scale ℓ and the latent scale σ_f determine the working regime (1-9) of the Gaussian Process as drafted in figure 4.3. We use the cumulative logistic likelihood and the squared exponential covariance function to classify handwritten digits. The four panels illustrate the model performance in terms of evidence, information and classification errors over the space of hyperparameters (ℓ, σ_f). For better visibility we choose a logarithmic scale of the axes. Panel a) shows the inherent evidence approximation of the four methods and panel b) contains the Jensen lower bound (equation 4.13) on the evidence used in KL method. Both panels share the same contour levels for all four methods. Note that for the VB method, the general lower bound is a better evidence estimate than the bound provided by the method itself. Panel c) and d) show the information score and the number of misclassifications.

One can read-off the divergence between posterior and approximation by recalling $KL(\mathbb{Q}||\mathbb{P}) = \ln Z - \ln Z_{KL}$ from equation 4.14 and assuming $\ln Z_{EP} \approx \ln Z$. In the figure this corresponds to subtracting subplots b, LA-VB) from subplots a, EP). Obviously, the divergence vanishes for close-to-Gaussian posteriors (regimes 3, 5-6, 7-9).

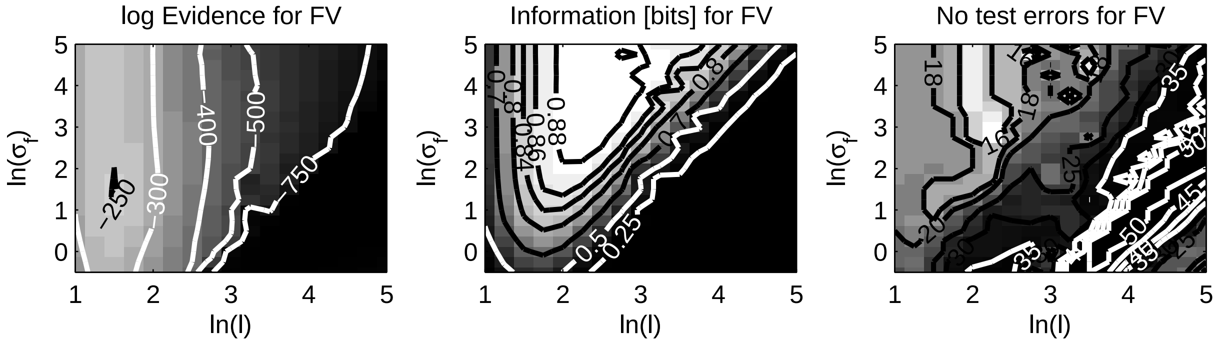
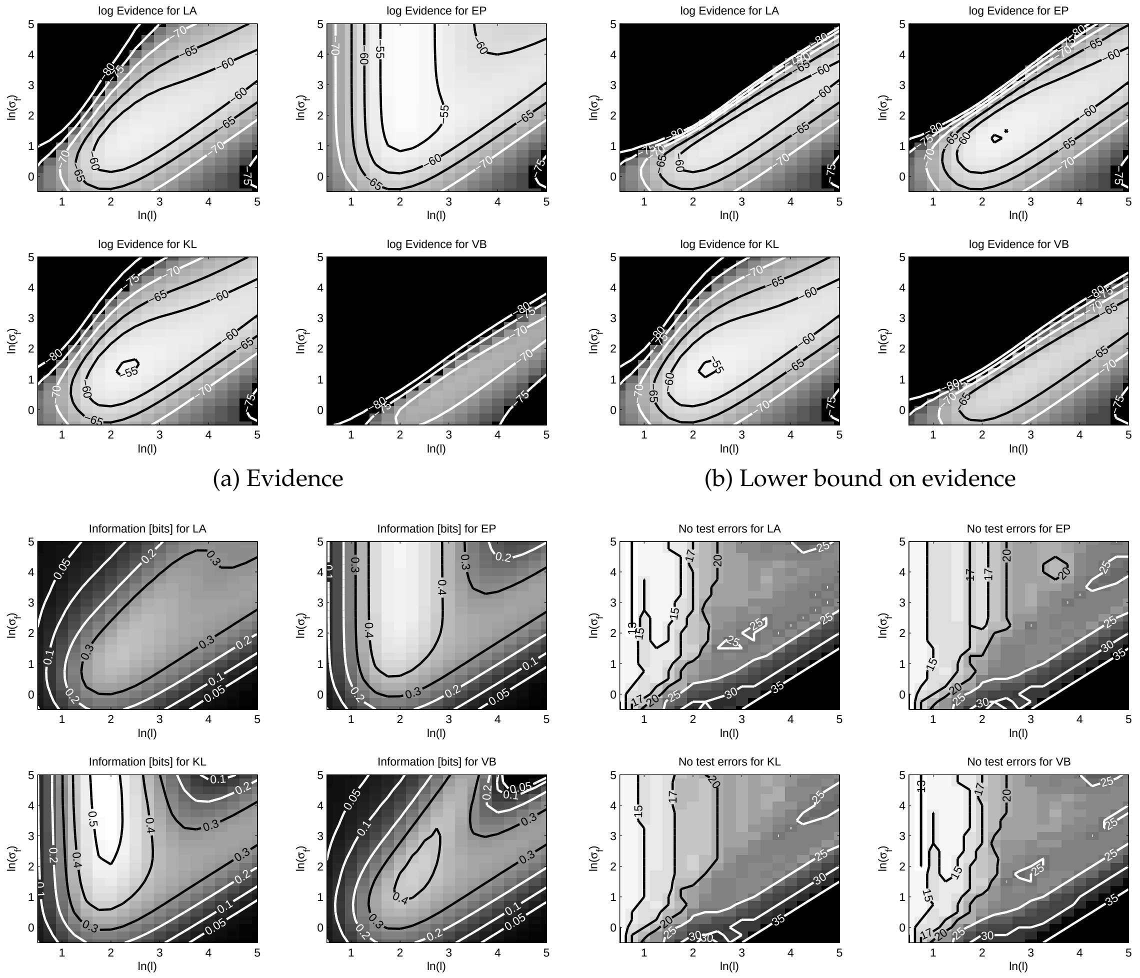


Figure 4.10: Evidence and classification performance for FV on USPS 3 vs. 5
The plots are a supplement to figure 4.9 making the factorial variational method comparable, even though we use the cumulative Gaussian likelihood. The levels of the contour lines for the information score and the number of misclassifications are the same as in figure 4.9. For the marginal likelihood other contours are shown, since it has significantly different values.



(a) Evidence

(b) Lower bound on evidence

(c) Information in bits

(d) Number of errors

Figure 4.11: Evidence and classification performance for LA, EP, KL & VB on sonar
We show the same quantities as in figure 4.9, only for the Sonar Mines versus Rocks dataset and using the cumulative Gaussian likelihood function.

4.12.0.2 Predictive probability p_* and information score I

Low-level features like posterior moments are not a goal per se, they are only needed for the purpose of calculating predictive probabilities. figures 4.4 and 4.6 show predictive probabilities in the last row.

In principle, a bad approximation in terms of posterior moments can still provide reasonable predictions. Consider the predictive probability from equation 4.23 using a cumulative Gaussian likelihood

$$p_* = \int \mathrm{sig}_{\mathrm{probit}}(f_*)\mathcal{N}(f_*|\mu_*, \sigma_*^2)\mathrm{d}f_* = \mathrm{sig}_{\mathrm{probit}}(\mu_*/\sqrt{1+\sigma_*^2}).$$

It is easy to see that the predictive probability p_* is constant if $\mu_*/\sqrt{1+\sigma_*^2}$ is constant. That means, moving mean μ_* and standard deviation σ_* along the hyperbolic curve $\mu_*^2/C^2 - \sigma_*^2 = 1$, while keeping the sign of μ_* fixed, does not affect the probabilistic prediction. In the limit of large μ_* and large σ_*, rescaling does not change the prediction.

Summarising all predictive probabilities p_i we consider the scaled information score I. As a baseline model we use the best model ignoring the inputs $\mathbf{x}_i$. This model simply returns predictions matching the class frequencies of the training set.

$$B = -\sum_{y=\{+1,-1\}} \frac{n_{\mathrm{test}}^{y}}{n_{\mathrm{test}}^{+1}+n_{\mathrm{test}}^{-1}} \log_2 \frac{n_{\mathrm{train}}^{y}}{n_{\mathrm{train}}^{+1}+n_{\mathrm{train}}^{-1}} \leq 1[\mathrm{bit}]$$

We take the difference between the baseline B (entropy) and the average negative log predictive probabilities $\log_2 \mathbb{P}\left(y_*|\mathbf{x}_*, \mathbf{y}, \mathbf{X}\right)$ to obtain the information score

$$I = B + \frac{1}{2n_{\mathrm{test}}} \sum_{i=1}^{n_{\mathrm{test}}} (1+y_i)\log_2(p_i) + (1-y_i)\log_2(1-p_i),$$

which is 1[bit] for perfect (and confident) prediction and 0[bit] for random guessing (for equiprobable classes). Figures 4.9c, 4.10(middle) and 4.11c contain information scores for 5 different approximation methods on two different datasets as a function of the hyperparameters of the covariance function. According to the EP and KL plots (most prominently in figure 4.11c), there are two strategies for a model to achieve good predictive performance:

- Find a good length scale ℓ (e.g. $\ln \ell \approx 2$) and choose a latent function scale σ_f above some threshold (e.g. $\ln \sigma_f > 3$).
- Start from a good set of hyperparameters (e.g. $\ln \ell \approx 2$, $\ln \sigma_f \approx 2$) and compensate a harder cutting likelihood ($\sigma_f^2 \uparrow$) by making the data points more similar to each other ($\ell^2 \uparrow$).

The LA method heavily underestimates the marginal means in the non-Gaussian regime (see regimes 1-5 of figure 4.3). As a consequence, the predictive probabilities are strongly underconfident in the non-Gaussian regime, first column of figure 4.6. The information score's value is too small in the non-Gaussian regime, figures 4.9c and 4.11c.

Since the EP algorithm yields marginal moments very close to the MCMC estimates (second column of figure 4.6), its predictive probabilities and information score is consequently also very accurate, figures 4.9c and 4.11c. The plots corresponding to EP can be seen as the quasi gold standard [Kuss and Rasmussen, 2005, figures 4 and 5].

The KL method slightly underestimates the variance and slightly overestimates the mean, which leads to slightly overconfident predictions, fourth column of figure 4.6. Overconfidence, in general, leads to a degradation of the information score, however in this example, the information score is very close to the EP values and at the peak it is even slightly (0.01[bit]) higher, figures 4.9c and 4.11c.

The VB method, again, has the same problems as the KL method only amplified. The predictions are overconfident, third column of figure 4.6. Consequently, the information measured score in the non-Gaussian regime is too small. The cumulative logistic likelihood function (figure 4.9c) yields much better results than the cumulative Gaussian likelihood function (figure 4.11c).

Finally, as the FV method is accurate if the prior is isotropic, predictive probabilities and information scores are very high in regimes 1, 4 and 7 of figure 4.3. For correlated priors, the FV method achieves only low information scores, figure 4.10(middle). The method seems to benefit from the "hyperbolic scaling invariance" of the predictive probabilities mentioned earlier in that section because both the mean and the variance are strongly underestimated.

4.12.0.3 Number of errors E

If there is only interest in the actual class and not in the associated confidence level, one can simply measure the number of misclassifications. Results for 5 approximation methods and 2 datasets are shown in figures 4.9d, 4.10(right) and 4.11d.

Interestingly, all four Gaussian approximation have very similar error rates. The reason is mainly due to the fact that all methods manage to compute the right sign of the marginal mean. Only the FV method with cumulative Gaussian likelihood seems a bit problematic, even though the difference is only very small. Small error rates do not imply high information scores, it is rather the other way round. In figure 4.9d at $\ln \ell = 2$ and $\ln \sigma_f = 4$ only 16 errors are made by the LA method while the information score (figure 4.9c) is only of 0.25[bits].

Even the FV method yields very accurate classes, having only small error rates.

4.12.0.4 Marginal likelihood Z

Agreement of model and data is typically measured by the marginal likelihood Z. Hyperparameters can conveniently be optimised using Z not least because the gradient $\frac{\partial \ln Z}{\partial \boldsymbol{\theta}}$ can be analytically and efficiently computed for all methods. Formally, the marginal likelihood is the volume of the product of prior and likelihood. In classification, the likelihood is a product of sigmoid functions (figure 4.3), so that only the orthant $\{\mathbf{f}|\mathbf{f} \odot \mathbf{y} \geq \mathbf{0} \in \mathbb{R}^n\}$ contains values $\mathbb{P}(\mathbf{f}|\mathbf{y}) \geq \frac{1}{2}$. In principle, evidences are bounded by $\ln Z \leq 0$, where $\ln Z = 0$ corresponds to a perfect model. As pointed out in section 4.2.0.1, the marginal likelihood for a model ignoring the data and having equiprobable targets has the value $\ln Z = -n \ln 2$, which serves as a baseline.

Evidences provided by LA, EP and VB for two datasets are shown in figures 4.9a, 4.10(left) and 4.11a. As the Jensen bound can be applied to any Gaussian approximation of the posterior, we also report it in figures 4.9b and 4.11b.

The LA method strongly underestimates the evidence in the non-Gaussian regime, because it is forced to centre its approximation at the mode, figures 4.9a and 4.11a. Nevertheless, there is a good agreement between the value of the marginal likelihood and the corresponding information score. The Jensen lower bound is not tight for the LA approximation, figures 4.9b and 4.11b.

The EP method yields the highest values among all other methods. As described in section 4.2.0.2, for high latent function scales σ_f^2, the model becomes effectively independent of σ_f^2. This behaviour is only to be seen for the EP method, figures 4.9a and 4.11a. Again, the Jensen bound is not tight for the EP method, figures 4.9b and 4.11b. The difference between EP and MCMC marginal likelihood estimate is vanishingly small [Kuss and Rasmussen, 2005, figures 4 and 5].

The KL method directly uses the Jensen bound (equation 4.12), which can only be tight for Gaussian posterior distributions. If the posterior is very skew, the bound inherently underestimates the marginal likelihood. Therefore, figures 4.9a and 4.9b and figures 4.11a and 4.11b show the same values. The disagreement between information score and marginal likelihood makes hyperparameter selection based on the KL method problematic.

The VB method's lower bound on the evidence turns out to be very loose, figures 4.9a and 4.11a. Theoretically, it cannot be better than the more general Jensen bound due to the additional constraints imposed by the individual bound on each likelihood factor, figures 4.9b and 4.11b. In practise, one uses the Jensen bound for hyperparameter selection. Again, the maximum of the bound to the evidence is not very helpful for finding regions of high information score.

Finally, the FV method only yields a poor approximation to the marginal likelihood due to the factorial approximation, figure 4.10. The more isotropic the model gets (small ℓ), the tighter is the bound. For strongly correlated priors (large ℓ) the evidence drops even below the baseline $\ln Z = -n \ln 2$. Thus, the bound is not adequate to do hyperparameter selection as its maximum does not lie in regions with high information score.

4.12.0.5 Choice of likelihood

In our experiments, we worked with two different likelihood functions, namely the cumulative logistic and the cumulative Gaussian likelihood. The two functions differ in their slope at the origin and their asymptotic behaviour. We did not find empirical evidence supporting the use of either likelihood. Theoretically, the cumulative Gaussian likelihood should be less robust against outliers due to the quadratic asymptotics. Practically, the different slopes result in a shift of the latent function length scale in the order of $\ln \frac{1}{4} - \ln \frac{1}{\sqrt{2\pi}} \approx 0.46$ on a log scale; the cumulative logistic likelihood prefers a bigger latent scale. Only for the VB method, differences were significant because the cumulative logistic bound is more concise.

Results across datasets

We conclude with a quantitative summary of experiments conducted on 6 datasets (breast, crabs, ionosphere, diabetes, sonar, USPS 3 vs. 5), two different likelihoods (cumulative Gaussian, cumulative logistic) and 8 covariance functions (linear, polynomial of degree 1-3, Matérn $\nu \in \{\frac{3}{2}, \frac{5}{2}\}$, squared exponential and neural network) resulting in 96 trials. All 7 approximate classification methods were trained on a 16×16 grid of hyperparameters to compare their behaviour under a wide range of conditions. We calculated the maximum (over the hyperparameter grid) amount of information, every algorithm managed to extract from the data in each of the 96 trials. Table 4.10 shows the number of trials, where the respective algorithm had a maximum information score that was above the mean/median (over the 7 methods).

Test \ Method	**LA**	**EP**	**KL**	**VB**	**FV**	**LR**	**TAPnaive**
# trials, information below **mean**	31	0	0	6	34	92	31
# trials, information below **median**	54	0	0	15	48	96	51

Table 4.10: Algorithm comparison across datasets

4.13 Discussion

We provide a comprehensive overview of methods for approximate Gaussian process classification. We present an exhaustive analysis of the considered algorithms using theoretical arguments. We deliver thorough empirical evidence supporting our insights revealing the strengths and weaknesses of the algorithms. Finally, we make a unified and modular implementation of all methods available to the research community.

We are able to conclude that the expectation propagation algorithm is, in terms of accuracy, always the method of choice, except if you cannot afford the slightly longer running time compared to the Laplace approximation.

Our comparisons include the Laplace approximation and the expectation propagation algorithm [Kuss and Rasmussen, 2005]. We extend the latter to the cumulative logistic likelihood. We apply Kullback-Leibler divergence minimisation to Gaussian process classification and derive an efficient Newton algorithm. Although the principles behind this method have been known for some time, we are unaware that this method has been previously implemented for GPs in practise. The existing variational method [Gibbs and MacKay, 2000, Jaakkola and Jordan, 1996] is extended by a lower bound on the cumulative Gaussian likelihood and we provide an implementation based on Newton's method. Furthermore, we give a detailed analysis of the factorial variational method [Csató et al., 2000].

All methods are considered in a common framework, approximation quality is assessed, predictive performance is measured and model selection is benchmarked.

In practise, an approximation method has to satisfy a wide range of requirements. If **runtime** is the major concern or one is interested in **error rate** only, the Laplace approximation or label regression should be considered. But only expectation propagation and – although a lot slower – the KL-method deliver accurate **marginals** as well as reliable **class probabilities** and allow for faithful **model selection**.

If an application demands a **non-standard likelihood** function, this also affects the choice of the algorithm: the Laplace application requires derivatives, expectation propagation and the factorial variational method need integrability with respect to Gaussian measures. However, the KL-method simply needs to evaluate the likelihood and known lower bounds naturally lead to the VB algorithm.

Finally, if the classification problem contains a lot of **label noise** (σ_f is small), the exact posterior distribution is effectively close to Gaussian. In that case, the choice of the approximation method is not crucial since in the Gaussian regime, they will give the same answer. For weakly coupled training data, the factorial variational method can lead to quite reasonable approximations.

As a future goal remains an in-depth understanding of the properties of sparse and online approximations to the posterior and a coverage of a broader range of covariance functions. Also, the approximation techniques discussed can be applied to other non-Gaussian inference problems besides the narrow applications to binary GP classification discussed here, and there is hope that some of the insights presented may be useful more generally.

Chapter 5

Adaptive Compressed Sensing of Natural Images

Multivariate real-world signals are highly structured: For example, the redundancy contained in natural images, e.g. sparsity after some linear transform, can be used for compression without perceptible loss. As a consequence, one can store an image much more efficiently than an unstructured collection of independent pixels. However, typical image acquisition devices such as digital cameras are not aware of this structure during the acquisition process: they measure every pixel independently. Only later when the image is *stored*, redundancy is exploited in compression schemes like JPEG.

Recently the research field of *compressed sensing* (CS) [Candès et al., 2006, Donoho, 2006a] with theoretical underpinnings from approximation theory [Ismagilov, 1974, Kashin, 1978, Garnaev and Gluskin, 1984] emerged. Its main goal is to exploit redundancy in the acquisition process already. The main result is that structured signals like images can be sampled below the Nyquist-limit and still be reconstructed to satisfaction, if nonlinear reconstruction algorithms are used and regular undersampling designs are avoided. The randomised measurement design, however, is non-adaptive to the particular signal to be measured itself.

In this chapter which is an extended version of Seeger and Nickisch [2008a], we address the CS problem within the general framework of statistical (Bayesian) experimental design. For particular natural images, we optimise the sub-Nyquist image measurement architecture so that the subsequently nonlinearly reconstructed image contains as much information as possible. We present experimental results shedding more light on how to make CS work for images. In a large study using 75 standard images, we compare various CS reconstruction methods utilising random measurement filters from different ensembles to a number of techniques which sequentially search for these filters, including our own, and Bayesian projection optimisation [Ji and Carin, 2007]. Similar to Weiss et al. [2007], we find that a simple heuristic of measuring wavelet coefficients in a fixed, top-down ordering significantly outperforms CS methods using random measurements, even if modern CS reconstruction algorithms are applied; the approach of Ji and Carin [2007] performs even worse. Beyond that, we show that our efficient approximation to sequential Bayesian design can be used to learn measurement filters which indeed outperform measuring wavelet coefficients top-down. Our results show that the property of incoherence of a measurement design, which plays a central role in the "unstructured except for random sparsity" theoretical CS setting, bears only little significance for measuring real natural images. As we will discuss in more detail, our findings indicate that certainly for natural images, but also for other signals with non-Gaussian but structured statistics, measurement designs can be optimised in a data-driven way from little concrete prior knowledge, with outcomes that can be significantly superior to uninformed or even coloured random designs. The main property driving the design optimisation in our case is the ability of the Bayesian reconstruction method to maintain valid uncertainty beliefs about its point estimates at all times.

The structure of the chapter is as follows. The experimental design approach to CS is introduced in section 5.1 and our image acquisition model is detailed in section 5.2. Our framework for approximate inference is described in section 5.3, where we also show how to apply it to large problems, especially for sequential acquisition. Other approaches to the same problem are reviewed in section 5.4. The empirical validation encompasses a series of experiments, comparing a range of adaptive compressed sensing methods on artificial data (section 5.5.1), and on the problem of measuring natural images (section 5.5.2).

5.1 Introduction

Compressed sensing [Candès et al., 2006, Donoho, 2006a], also known as compressive sampling, can be motivated as follows. Suppose a signal, such as an image or a sound waveform, is measured and then transferred over some channel or stored. Traditionally, the measurement obeys the Nyquist theorem, allowing for an exact reconstruction of any (band-limited) signal. However, what follows is usually some form of lossy compression, exploiting redundancies and non-perceptibility of losses. Given that, can the information needed for a satisfactory reconstruction not be measured below the Nyquist frequency by so called *undersampling*? In many key applications today, the measurement itself is the main bottleneck for cost reductions or higher temporal/spatial resolution. Recent theoretical results indicate that undersampling should work well if *randomised designs* are used, and if the signal reconstruction method specifically takes the *compressibility* into account.

We formally introduce redundancy in section 5.1.1, then define the CS problem and describe in section 5.1.2 how experimental design can be used to tackle it and finally discuss adaptive compressed sensing in section 5.1.3.

5.1.1 Redundancy, compressibility and natural images

Intuitively, redundancy is equivalent to compressibility of a signal since the two terms mutually imply each other. Formally, Shannon's source coding theorem [Shannon, 1948] states that the minimal per-variable code length of an infinitely long sequence of (i.i.d.) random variables $\mathbf{x}_i \sim \mathbb{P}(\mathbf{x})$ is precisely given by the entropy $\mathcal{H}[\mathbb{P}(\mathbf{x})]$. For fixed mean and variance, the multivariate Gaussian distribution has maximal entropy making Gaussian noise the least structured signal with maximal coding length. For a multivariate random variable, entropy depends not only on non-Gaussianity but also on mutual dependencies. Firstly, independence relations increase entropy

$$\mathcal{H}[\mathbb{P}(x_i, x_j)] = \mathcal{H}[\mathbb{P}(x_i)] + \mathcal{H}[\mathbb{P}(x_j)] - \mathcal{I}(x_i, x_j) \leq \mathcal{H}[\mathbb{P}(x_i)] + \mathcal{H}[\mathbb{P}(x_j)],$$

i.e. the joint entropy $\mathcal{H}[\mathbb{P}(x_i, x_j)]$ is maximal if x_i, x_j are independent, which means they have mutual information $\mathcal{I}(x_i, x_j)$ zero. Secondly, Gaussianity increases entropy (see appendix D.4)

$$\mathcal{H}[\mathbb{P}(x_i)] \leq \mathcal{H}[\mathcal{N}(x_i|\mu_i, \sigma_i^2)],\ \mu_i = \mathbb{E}[x_i],\ \sigma_i^2 = \mathbb{V}[x_i]$$

meaning that non-Gaussian distributions allow for better compression[1]. Natural images show both: super-Gaussian marginals in the gradient domain called sparsity and strong pixel covariance also referred to as second order structure.

Most of the theoretical work on CS however, considers the asymptotic minimax performance of certain penalised estimators. In general, signals are assumed to be *unstructured except for random sparsity* – a concept whose validity depends on whether we aim to hedge against the worst case, or whether we place ourselves in a more benevolent setting, where active reductions in uncertainty normally lead to better predictions.

Natural images exhibit transform sparsity, yet random measurements favoured by CS theory can be suboptimal for them [Weiss et al., 2007]. The reason is that there is – as pointed

[1] Of course, continuous random variables have to be discretised to be stored on a computer. If discretised into equal bins $\mathcal{B}_i = b_0 + [i-1, i] \cdot \Delta$, $i = 1..N$ (minimises maximum democratisation error), the entropy code uses code words of lengths $\ell_i = \log_N p_i$, where $p_i = \int_{\mathcal{B}_i} \mathbb{P}(\mathbf{x})\mathrm{d}\mathbf{x}$ is the probability of the ith symbol.

before – more to low-level image statistics than random sparsity alone; knowledge that can be modelled tractably [Simoncelli, 1999].

5.1.2 The compressed sensing problem and experimental design

It is important to distinguish between i) the CS problem, ii) signal characteristics making CS possible, iii) reconstruction methods incorporating these properties, and iv) theoretical results about the problem in principle, or v) about specific reconstruction methods. In the recent surge of activity on CS, such distinctions are not always precisely stated, which may lead to confusion. CS constitutes a *problem*, which in practise is amply motivated by cost reductions. Fewer measurements, or less precise sampling, can lead to similar quality in signal reconstruction, at the expense of having to design and run a more difficult reconstruction method, and also (in general) of having to modify "standard" measurement designs. Not all types of signals are admissible to CS. For example, for band-limited random noise, the Nyquist theorem is tight. In general, CS is applicable to signals whose distribution has some structure that is known *a priori*, before any measurements are done. Since such knowledge can be used to compress samples, signals of that sort are also called *compressible*. A very important structure, which is characteristic to some extent for many signals, is *sparsity*: if the signal in its standard representation is transformed linearly, most coefficients are very close to zero, while a few can be large. We will discuss sparsity below in more detail. One can think about structural prior knowledge as a (partial) ordering on the representation space of the signal. In this ordering, a signal is "less complex" than another one, if it adheres better to prior knowledge.

Any *solution* to the CS problem has to master two related, but different tasks. First, for given measurements, an estimate of the signal has to be computed taking into account prior knowledge. This is called *signal reconstruction*. Second, the decision of how to measure in the first place has to be taken.

Bayesian experimental design offers a powerful way of addressing both points. The structural prior knowledge about a signal (its compressibility) is encoded into a *prior distribution*, under which signals of low complexity in general, or high (transform) sparsity in particular, have most mass. By the Nyquist theorem, *all* signals within some band are identifiable through the *likelihood function* of measurements spaced closely enough. A Bayesian (as well as a CS) reconstruction of the signal, however, is obtained by *combining likelihood and prior*: signals which are sufficiently likely under the prior, can often be reconstructed from a likelihood function of undersampled measurements[2], at lower cost than with a foolproof Nyquist-spaced sample.

The problem of optimising the measurement structure (or *design*), so that less measurements are needed to attain the same reconstruction quality, is harder in general. For this problem, Bayesian *experimental design* offers a powerful and general solution. In the context of natural images, maximally incoherent (random) designs perform rather poorly, while properly optimised designs can improve upon the engineering status quo. Remarkably, the *same* prior knowledge is available to both Bayesian design and CS reconstruction methods. While in our Bayesian setup, prior and observations are used in order to choose good subsequent measurements, this seems hard to do with CS point estimation techniques.

5.1.3 Adaptive sequential compressed sensing

In order for CS to work, one exploits compressibility properties of a general class of signals. However, it is clear, that one can improve by restricting the signal class. An acquisition process depending on the particular signal one is measuring, is called *adaptive*. Furthermore, if the next acquisition step depends on previous ones, the acquisition is termed *sequential*.

Our setup is sequential; new measurements are appended to the measurement design one at a time. Adaptive techniques, such as ours, make use of all measurements obtained so far

[2] The Nyquist theorem states that there are always *some* signals that cannot be reconstructed properly from an undersampled likelihood, but a well-chosen design can ensure that most of these "bad signals" have very low prior probability.

to decide upon the next, while *non-adaptive* methods neglect this sequential information. A simple non-adaptive approach is to sample the design matrix at random, using independent Gaussian or Bernoulli entries, or random rows of the *discrete Fourier transform* (DFT) matrix. Also, coloured random projections have been proposed [Wang et al., 2007], to take into account second order structure of the signal besides sparsity. A different approach for *a priori* measurement design is given in Elad [2007], where the measurement matrix is optimised to make its rows maximally incoherent with the sparsifying transform. A similar argument lets Candès et al. [2006] use the noiselet transform [Coifman et al., 2001]: it is maximally incoherent to the Haar wavelet basis.

On the other hand, with adaptive techniques, the next measurement is chosen to maximise a criterion which depends on the measurements made so far. For example, the hierarchical nature of multi-scale wavelet coefficients motivates the adaptive heuristic proposed in Dekel [2008]. An approximate *Bayesian* approach to *compressed sensing* (BCS) has been presented in Ji and Carin [2007], making use of *sparse Bayesian learning* (SBL) [Tipping, 2001]. The method can be improved by exploiting the structure in the wavelet transform [He and Carin, 2009].

We extend the BCS/SBL approach by using a more general inference approximation, expectation propagation [Minka, 2001a], leading to much better reconstruction performance in our application. As we argue below, BCS/SBL method seems to be over-aggressive in terms of sparsification, leading to avoidable mistakes on natural images, which are just not strictly sparse in general. Moreover, their uncertainty (posterior covariance) estimates seem to be adversely affected by the aggressiveness, which in turn spoils design adaptation. In addition, our framework is easily generalised to non-Gaussian observation likelihoods, skew prior terms, and generalised linear models [Gerwinn et al., 2008], and our methodology and comparisons have a broader scope. In the next section, we will describe the probabilistic model in detail.

5.2 Probabilistic natural image acquisition

Bayesian experimental design (see chapter 2.6.2) for optimising natural image acquisition fits into the linear model framework of chapter 2. Here, an image is represented as a pixelised bitmap, which (for notational convenience only) is stacked into a vector $\mathbf{u} \in \mathbb{R}^n$ (where n is the number of pixels). In our example, u_i are grey-scale values, but an extension to colour images is straightforward. The task is to reconstruct $\mathbf{u}$ (the *latent variables*) from noisy linear measurements

$$\mathbf{y} = \mathbf{X}\mathbf{u} + \boldsymbol{\varepsilon}, \quad \mathbf{X} \in \mathbb{R}^{m \times n}, \ \boldsymbol{\varepsilon} \sim \mathcal{N}(\mathbf{0}, \sigma^2\mathbf{I}). \tag{5.1}$$

$\mathbf{X}$ is called the *design* or *measurement matrix*, its rows are *measurement filters*. The filters are constrained to have unit norm[3]. Note that $m < n$ in general, since measuring each pixel in turn is not considered an efficient design. The reconstruction problem is therefore *underdetermined*, and $(\mathbf{X}, \mathbf{y})$ constitute an undersampling of $\mathbf{u}$. The task is to choose the filters in a sequential manner (one after the other) to obtain a satisfactory reconstruction of $\mathbf{u}$ with as small m as possible. Note that in real-world instances of this problem, additional constraints on the filters (beyond unit norm) may be present. Our solution presented here readily extends to constrained filter optimisation as well (see section 5.4.2).

The prior distribution $\mathbb{P}(\mathbf{u})$ should encode properties which are characteristic of natural images, and this is where sparsity comes into play. While classical Bayesian analysis for the linear model (equation 5.1) employs Gaussian priors for $\mathbf{u}$, and experimental design is well-developed in general for the Gaussian case (see Chaloner and Verdinelli, 1995 and chapter 2.6.1), natural image statistics are distinctively non-Gaussian because zero mean filter responses of natural images follow sparse distributions [Simoncelli, 1999]. Our image prior here is composed of *Laplace* (or double exponential) potentials

$$\mathcal{T}_i(s_i) := \frac{\tau_i}{2} e^{-\tau_i |s_i|}, \quad s_i = [\mathbf{B}\mathbf{u}]_i = \mathbf{b}_i^\top \mathbf{u}, \tag{5.2}$$

[3] If we design $\mathbf{X}$, it will be important to keep its rows of the same scale. Otherwise, a measurement can always be improved (at fixed noise level σ^2) simply by increasing its norm.

whose coefficients s_i are linear functions of the image $\mathbf{u}$, collected in the transform matrix $\mathbf{B}$. In contrast to the Gaussian, the Laplacian is a sparsity-enforcing distribution: it concentrates more mass close to zero, but also has heavier tails. If $\mathbb{P}(\mathbf{u}) \propto \prod_i \mathcal{T}_i(s_i)$, then with Laplace potentials, the preference is for $\mathbf{s}$ to have most components very close to zero, allowing some components to be large, while with Gaussian potentials $\mathcal{T}_i$, no large s_i are tolerated, while there is also no pressure on the components to become very small. This notion is explained in more detail in Seeger [2008], Tipping [2001]. Our image prior employed here puts sparse distributions on multi-scale finite pixel differences. It falls naturally into two parts:

First, the *total variation* (TV) potential is a product of Laplace terms looking at image gradients by the extremely sparse finite difference matrix $\mathbf{D} \in \{-1, 0, +1\}^{2(n-\sqrt{n})\times n}$ so that $\mathbf{Du} = [\mathbf{d}_x; \mathbf{d}_y]$, with $\mathbf{d}_x, \mathbf{d}_y$ denoting the finite image derivatives in horizontal and vertical direction. The total variation potential can be written as $\exp(-\tau_D \|\mathbf{Du}\|_1)$, where $\|\mathbf{s}\|_1 := \sum_j |s_j|$ denotes the L_1 norm. They encode smoothness of images: neighbouring pixels tend to have similar grey-scale values, with occasional large differences due to edges, which agrees with the concentration at zero and the heavy tails of the Laplace density.

Second, the *wavelet* or *transform sparsity* potential looks at coarser scale derivatives as computed by the (orthonormal) wavelet transform $\mathbf{W}$ yielding $\exp(-\tau_W \|\mathbf{Wu}\|_1)$. Note that histograms of wavelet coefficients over natural images can be fit closely by a Laplace distribution [Simoncelli, 1999]. In our experiments, we always use the Daubechies 4 wavelet [Daubechies, 1992].

The parameters τ_D, τ_W represent the strength (or scale) of each term. Large values of τ_D, τ_W mean very tight potentials allowing only for small deviations from zero. $\mathbb{P}(\mathbf{u})$ is the normalised product of the two potentials[4]. Both matrices $\mathbf{D}$ and $\mathbf{W}$ are highly structured allowing for efficient matrix vector multiplications in $\mathcal{O}(n)$ time and space. Setting $\mathbf{B} = [\mathbf{D}; \mathbf{W}]$, our setup becomes an instance of the sparse linear model (SLM), where the Bayesian *posterior distribution* has the form

$$\mathbb{P}(\mathbf{u}|\mathbf{y}) \propto \mathcal{N}(\mathbf{y}|\mathbf{Xu}, \sigma^2\mathbf{I}) \prod_{i=1}^{q} \mathcal{T}_i(s_i), \quad \mathbf{s} = \mathbf{Bu}. \tag{5.3}$$

For large numbers of image pixels n, it is essential that matrix-vector multiplications (MVMs) with $\mathbf{X}, \mathbf{X}^\top$ can be computed efficiently, as well. Our framework can readily be used with $\mathcal{T}_i(s_i)$ that are not Laplace. If the $\mathcal{T}_i$ are log-concave, as is the case here, our method can be shown to be numerically stable [Seeger, 2008].

Many CS reconstruction methods (section 5.4) can be understood as *maximum a-posteriori* (MAP) estimation

$$\hat{\mathbf{u}} = \arg\max_{\mathbf{u}} \log \mathbb{P}(\mathbf{u}|\mathbf{y}) = \arg\max_{\mathbf{u}} \log \mathbb{P}(\mathbf{y}|\mathbf{u})\mathbb{P}(\mathbf{u}). \tag{5.4}$$

Here, $-\log \mathbb{P}(\mathbf{y}|\mathbf{u})\mathbb{P}(\mathbf{u})$ is referred to as *energy*, and MAP estimation as *energy minimisation*. If $-\log \mathbb{P}(\mathbf{y}|\mathbf{u})$ and $-\log \mathbb{P}(\mathbf{u})$ are convex in $\mathbf{u}$, as is the case for Gaussian and Laplace distributions, MAP estimation is a convex problem and can be solved efficiently. In this sense, the image prior constructed above is used in several CS estimation applications [Candès and Romberg, 2004], which is the main reason for using it here as well. In contrast, the Bayesian estimate of $\mathbf{u}$ is given by the posterior mean $\mathbb{E}[\mathbf{u}|\mathbf{y}] = \mathbb{E}_{\mathbb{P}(\mathbf{u}|\mathbf{y})}[\mathbf{u}]$. Decision theory (see chapter 2.1.2) states that the posterior mean is a better estimate than the posterior mode, if the objective is to minimise the squared error [Lehmann and Casella, 1998, chapter 4]. The mean is consistent under marginalisation (meaning that the Bayesian estimate of a part of the image is simply the corresponding part of the mean), while the mode is not. On the other hand, for the model considered here, no computationally tractable method for computing the exact mean is known

[4] $\mathbb{P}(\mathbf{u})$ is normalisable, because the transform sparsity potential is. Technically, the total variation potential is not normalisable on its own. However, it is still possible (and, in fact, works well) to use our method with $\tau_{sp} = 0$, since in undirected graphical models, the "prior" $\mathbb{P}(\mathbf{u})$ need not be normalisable. In general, $\mathbb{P}(\mathbf{u})$ should not be understood as a sensible generator for natural images anyway, but rather as incorporating *some* important natural image characteristics.

(even though $-\log \mathbb{P}(\mathbf{u}|\mathbf{y})$ is convex), and an approximation is harder to compute than solving for the mode (see section 5.3).

The problem of experimental design is to choose $\mathbf{X}$ among many candidates, so that subsequent measurements allow the best reconstruction of $\mathbf{u}$. Importantly, the approach is at least partly "closed-loop", in that it is not required to in fact *do* real measurements for most of the candidates. To understand this, keep in mind that (5.3) is only a *model* of the true measurement process, which however, combined with a growing number of real measurements, can successfully be used to predict the informativeness of new sampling not yet done. To do this, we need a *quantitative* statement about our uncertainty in $\mathbf{u}$ at the moment, which is the posterior $\mathbb{P}(\mathbf{u}|\mathbf{y})$. An extension of our design means new rows in $\mathbf{X}$. Its informativeness is scored by imagining the new measurement being done with outcome y_*, then measuring the *decrease in uncertainty* from $\mathbb{P}(\mathbf{u}|\mathbf{y})$ to $\mathbb{P}(\mathbf{u}|\mathbf{y}, y_*)$ as measured by the *entropy difference* or information gain (see chapter 2.6.2) $\mathcal{H}[\mathbb{P}(\mathbf{u}|\mathbf{y})] - \mathcal{H}[\mathbb{P}(\mathbf{u}|\mathbf{y}, y_*)]$. Since y_* is not known, it is integrated out using $\mathbb{P}(y_*|\mathbf{y}) = \int \mathbb{P}(y_*|\mathbf{u})\mathbb{P}(\mathbf{u}|\mathbf{y})\, d\mathbf{u}$. We now have information scores as criteria driving an optimisation of the design. It is clear that these are fundamentally based on a representation of uncertainty, the posterior in the Bayesian case, and that algorithms which merely estimate point solutions from given data, cannot be used directly in order to compute them. With such methods, either rules of thumb have to be followed to obtain a design (such as "do it at random"), or many measurements have to be taken in a trial-and-error fashion. The edge of Bayesian experimental design is that through a combination of the model and real measurements, a continuously refined uncertainty statement is obtained, based on which uninformative sampling can often be avoided. This way, often substantially fewer real measurements are required. Another important point is that experimental design works, although the true underlying $\mathbf{u}$ is not known. This is what drives *sequential* design optimisation and makes the gathering of large "training data" collections unnecessary[5].

5.3 Approximate inference

Bayesian inference is in general not analytically tractable for models of the form (5.3), and has to be approximated. Moreover, the application of interest here demands high efficiency in many dimensions ($n = 4096$ in the natural image experiments here). Importantly, Bayesian experimental design does not only require inference just once, but many times in a sequential fashion. We make use of the *expectation propagation* (EP) method [Minka, 2001a], together with a robust and efficient representation for $\mathbb{Q}(\mathbf{u}) \approx \mathbb{P}(\mathbf{u}|\mathbf{y})$. As a novelty, we will show here how the framework can be run efficiently for large n, and how sequential design optimisation can be sped up by orders of magnitude.

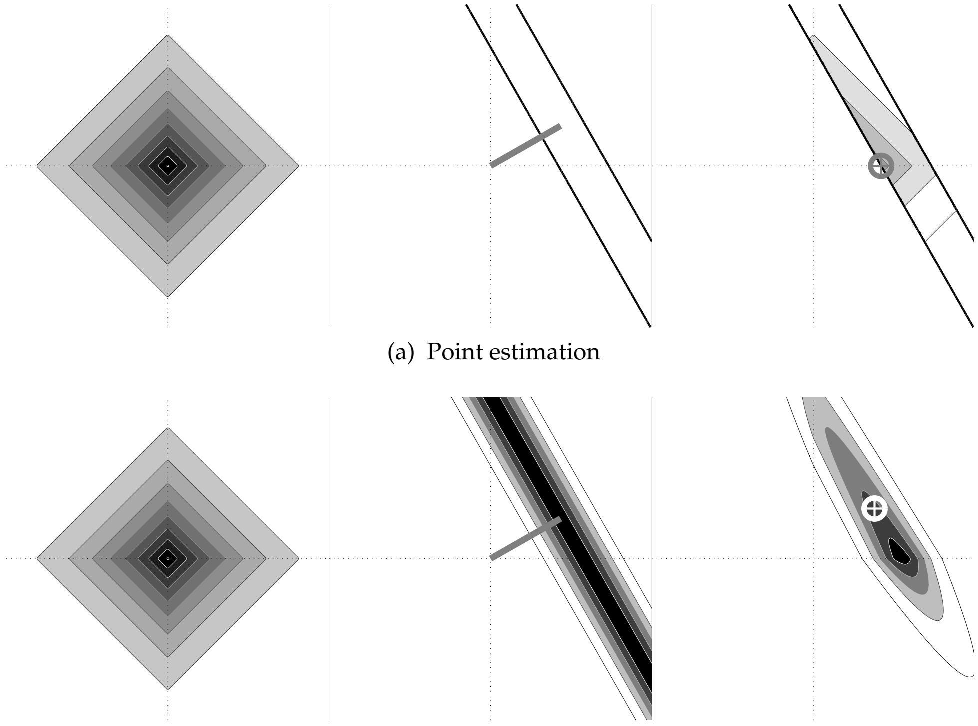
We first provide some intuition about the inference method in terms of what it is going to achieve and also in terms of the underlying geometry. Then, we will discuss the technical formulation of the algorithm and comment on how to scale it up to large sizes.

5.3.1 Inference and estimation

Before we describe the EP approximation, we will give an intuitive view on what inference is about, and how algorithms to approximate it differ from estimation methods. In many statistical problems — certainly the ones concerned with images — experience suggests that there are many potential constraints, which should to some degree be met by the underlying signal to be reconstructed. For example, observations imply constraints through likelihood terms, each of which may depend on all latent variables. Moreover, prior constraints for images are often local in nature, enforcing smoothness by constraining neighbouring pixels to have similar values, as in the total variation potential described above. However, strictly enforcing *all* constraints is usually not possible, or leads to trivial solutions. Rather, the constraints have to

[5] Another way to view experimental design is that this process of gathering training data is done actively, so that data is sampled where really needed to gain further clarity, typically at substantial reductions in cost.

be weighted against each other. In estimation methods, this constraint weighting is done in a rough way: either, some constraints have to be met (infinite weight), or the constraints are split into two groups (usually likelihood versus prior), with equal weighting within groups (see section 5.4). In contrast, with Bayesian inference, all constraints are fundamentally probabilistic. An approximate inference method such as EP can be thought of as finding a proper weighting across all constraints in an iterative process of negotiation between all model potentials: “messages” are exchanged between neighbouring potentials, until at convergence an equilibrium of mutual agreement is established. Importantly for our application here, these negotiation mechanisms are in terms of distributions (or beliefs), encoding *uncertainties* of potentials about the state of neighbouring ones or about their own state. At convergence, these beliefs approximate posterior uncertainties, which in turn drive Bayesian experimental design. Moreover, we will see below how they can be used within the algorithm itself, in order to attain faster convergence. These additional information sources are not required, and therefore not present, in pure estimation methods.



(a) Point estimation

(b) Bayesian posterior mean and MAP point estimation

Figure 5.1: Geometrical illustration of several inference and estimation methods
We geometrically contrast the penalised least squares estimator with the posterior mean and mode estimator.
In panel 5.1a), we depict point estimation in the sparse linear model. From left to right: sparsity objective $\|\mathbf{u}\|_1$, feasible region $\{\mathbf{u} \in \mathbb{R}^n \mid \frac{1}{2}\|\mathbf{X}\mathbf{u}-\mathbf{y}\|_2 \le \sigma\}$ ($\mathbf{X} \equiv$ grey line, $\mathbf{B} = \mathbf{I}$), optimal solution (grey cross). Note that the estimator is sparse since the optimum will be at a corner, here $\hat{u}_2 = 0$.
Panel 5.1b) illustrates Bayesian inference. From left to right: sparsity prior $\prod_i \mathcal{T}_i(u_i|\tau)$, observation likelihood $\mathcal{N}(\mathbf{y}|\mathbf{X}\mathbf{u},\sigma^2\mathbf{I})$ ($\mathbf{X} \equiv$ grey line, $\mathbf{B} = \mathbf{I}$), posterior distribution $\mathbb{P}(\mathbf{u}|\mathbf{y})$ and its mean (white cross). The MAP or mode estimator is found at the (black) peak of the posterior. Note that the MAP estimator also exhibits sparsity.

Pictorial geometrical illustration

Figure 5.1 provides an (admittedly low-dimensional) geometrical intuition about the relations between different estimation techniques. Figure 5.1a illustrates the situation for the relaxed L_1 case: the L_1 regulariser is minimised inside the feasible region – the estimator chooses among all feasible coefficients the ones with maximal sparsity. Many of the coefficients of the solution will turn out to be zero, since the optimum is attained at a corner of the objective. The Bayesian inference case is shown by figure 5.1b: the sparsity prior assigns higher probabilities to signals close to the coordinate axes. The likelihood smoothly cuts out the subspace compatible with the noisy observations. Combining both of them, the posterior puts mass to all plausible signals under our model. The posterior mode also shows sparsity characteristics. The posterior mean is the best signal estimate in the squared error sense [Lehmann and Casella, 1998]. Since EP is the most accurate way (see experiments in chapter 4) of approximately computing posterior moments such as the mean, we choose it as our inference engine.

5.3.2 Expectation propagation

In EP (see also chapter 2.5.10), the posterior $\mathbb{P}(\mathbf{u}|\mathbf{y})$ is approximated by a Gaussian distribution $\mathbb{Q}(\mathbf{u})$ with $2q$ free (variational) parameters $(\boldsymbol{\beta},\boldsymbol{\gamma})$, which are formally introduced by replacing the non-Gaussian potentials $\mathcal{T}_i(s_i)$ by Gaussian potentials $\tilde{\mathcal{T}}_i(s_i) := e^{\beta_i s_i/\sigma^2 - s_i^2/(2\sigma^2\gamma_i))}$ in (equation 5.3). Beyond $(\boldsymbol{\beta},\boldsymbol{\gamma})$, it is usually necessary to maintain a *representation* of $\mathbb{Q}$, so that marginal distributions $\mathbb{Q}(s_i)$ can be obtained rapidly. For an EP update at potential i, we compute the Gaussian moments of the *tilted distributions*

$$\hat{\mathbb{P}}_i(\mathbf{u}) \propto \mathcal{N}(\mathbf{y}|\mathbf{X}\mathbf{u},\sigma^2\mathbf{I}) \prod_{j\neq i} \tilde{\mathcal{T}}_j(s_j)\tilde{\mathcal{T}}_i(s_i)^{1-\eta}\mathcal{T}_i(s_i)^{\eta},$$

then update $\mathbb{Q}(\mathbf{u})$ to match these moments, which can be done by modifying (β_i,γ_i) only. Here, $\eta \in (0,1]$ is a fractional parameter[6]. As motivated above, the single updates form a process of negotiation between all potentials $\mathcal{T}_i(s_i)$, which is resolved at convergence, where the means and covariances of all $\hat{\mathbb{P}}_i$ are the same. In each EP update, we merely need to compute mean and variance of the non-Gaussian *marginal* $\hat{\mathbb{P}}_i(s_i)$, and to update the $\mathbb{Q}(\mathbf{u})$ representation to accommodate the novel (β_i,γ_i) as detailed in the next section.

5.3.2.1 Posterior representation and update

A numerically stable representation of $\mathbb{Q}(\mathbf{u})$ [Seeger, 2008] maintains the $n \times n$ Cholesky factor $\mathbf{L}$ and the n vector $\boldsymbol{\alpha}$, so that

$$\begin{aligned}\mathbf{L}\mathbf{L}^\top &= \mathbf{X}^\top\mathbf{X} + \mathbf{B}^\top\boldsymbol{\Gamma}^{-1}\mathbf{B} = \sigma^2\left(\mathbb{V}_{\mathbb{Q}}[\mathbf{u}]\right)^{-1},\\ \boldsymbol{\alpha} &= \mathbf{L}^{-1}(\mathbf{X}^\top\mathbf{y} + \mathbf{B}^\top\boldsymbol{\beta}) = \mathbf{L}^\top\mathbb{E}_{\mathbb{Q}}[\mathbf{u}],\ \boldsymbol{\Gamma} = \mathrm{dg}(\boldsymbol{\gamma}).\end{aligned}$$

For an EP update at potential $\mathcal{T}_i$, we require $\mathbb{Q}(s_i) = \mathcal{N}(s_i|h_i,\sigma^2\rho_i)$, where $h_i = \mathbf{r}_i^\top\boldsymbol{\alpha}$, $\rho_i = \|\mathbf{r}_i\|^2$ with $\mathbf{r}_i = \mathbf{L}^{-1}\mathbf{b}_i$. The back-substitution costs $\mathcal{O}(n^2)$. The update requires finding β_i', γ_i', so that $\hat{\mathbb{P}}_i(s_i)$ and $\mathbb{Q}'(s_i)$ have the same mean and variance. Numerically stable moment matching is a nontrivial task. Finally, $\mathbf{L}$, $\boldsymbol{\alpha}$ are updated, using numerical mathematics code for rank one Cholesky update/downdate, which costs $\mathcal{O}(n^2)$.

5.3.2.2 Selective update and design

For selective potential updating, all marginals $(\mathbf{h},\boldsymbol{\rho})$ need to be present at all times (see section 5.3.3). This can be done by using the Woodbury formula at the cost of two back-substitutions with $\mathbf{L}$, rather than one only as detailed in Seeger [2008].

In our sequential design applications, score the informativeness of new candidates $\mathbf{x}_*$ (as potential new row of $\mathbf{X}$) by the entropy difference (see section 5.1). If $\mathbb{Q}'$ is the approximate posterior after including $\mathbf{x}_*$, then $\mathcal{H}[\mathbb{Q}'] = \log|\mathbb{V}_{\mathbb{Q}'}[\mathbf{u}]|/2 + C$, where $\mathbb{Q}'$ differs from $\mathbb{Q}$ in that

[6] $\eta = 1$ gives standard EP, but choosing $\eta < 1$ can increase the robustness of the algorithm on the sparse linear model significantly [Seeger, 2008]. We use $\eta = 0.9$ in all our experiments.

$(\mathbf{X}')^\top\mathbf{X}' = \mathbf{X}^\top\mathbf{X} + \mathbf{x}_*\mathbf{x}_*^\top$, and $\gamma \to \gamma'$. We approximate the entropy difference by assuming that $\gamma' = \gamma$, whence

$$\mathcal{H}[\mathbb{Q}] - \mathcal{H}[\mathbb{Q}'] = \frac{1}{2}\log\left(1 + \sigma^{-2}\mathbf{x}_*^\top \mathbb{V}_{\mathbb{Q}}[\mathbf{u}]\mathbf{x}_*\right).$$

Since $\|\mathbf{x}_*\|_2 = 1$ by assumption, this score is maximised by choosing $\mathbf{x}_*$ along the principal (leading) eigendirection of $\mathbb{V}_{\mathbb{Q}}[\mathbf{u}]$, which can be calculated by the Lanczos method [Lanczos, 1950, Golub and van Loan, 1996]. The same score is used in Ji and Carin [2007], yet the approximation of the posterior and its covariance is fundamentally different (see section 5.4).

5.3.3 Large scale applications

There will be two major issues if we apply our method for large image sizes n. First, the EP potential updates are typically done in random sweeps over all potentials, because it is not clear *a priori* which particular potential ordering leads to fastest convergence. This problem is severe in our sequential design application to natural images, since there are many small changes to $\mathbf{X}$, $\mathbf{y}$ (individual new measurements), after each of which EP convergence has to be regained. We approach it by forward scoring many potential candidates before each EP update, thereby always updating the one which gives the largest posterior change. This is detailed just below. Second, the robust $\mathbb{Q}$ representation of section 5.3.2.1, which is used in the experiments here, requires $\mathcal{O}(n^2)$ memory, and each update costs $\mathcal{O}(n^2)$ (see section 5.3.2.1). If $m \ll n$ at all times, a different representation of size $\mathcal{O}(m^2)$ can be used. Beyond that, our method can also be run representation-free, requiring $\mathcal{O}(n)$ storage only, if marginals are approximated by linear conjugate gradients and the Lanczos algorithm. However, either of these modifications leads to a loss in numerical accuracy.

Our selective updating scheme for EP hinges on the fact that we can maintain all potential marginals $(\mathbf{h}, \boldsymbol{\rho})$, $\mathbb{Q}(s_i) = \mathcal{N}(s_i|h_i, \sigma^2\rho_i)$, up-to-date at all times. We can quantify the change of $\mathbb{Q}$ through an update at a potential $\mathcal{T}_i$, by the relative entropy $\mathrm{KL}[\mathbb{Q}_i'(s_i)\,\|\,\mathbb{Q}(s_i)]$ ($\mathbb{Q}_i'$ the posterior after the update at $\mathcal{T}_i$), which can be computed in $\mathcal{O}(1)$. Here, the Kullback-Leibler divergence $\mathrm{KL}[\mathbb{P}\,\|\,\mathbb{Q}]$ measures the gain in information from $\mathbb{Q} \to \mathbb{P}$. Importantly, $\mathrm{KL}[\mathbb{Q}_i'(\mathbf{u})\,\|\,\mathbb{Q}(\mathbf{u})] = \mathrm{KL}[\mathbb{Q}_i'(s_i)\,\|\,\mathbb{Q}(s_i)]$, so the score precisely measures the global amount of change $\mathbb{Q} \to \mathbb{Q}_i'$. We maintain a list of candidate potentials which are scored before each EP update, and the update is done for the winner only. The list is then evolved by replacing the lower half of worst-scoring potentials by others randomly drawn from $\{1, .., q\}$. Importantly, the marginals $(\mathbf{h}, \boldsymbol{\rho})$ can be updated along with the representation of $\mathbb{Q}(\mathbf{u})$.

Our sequential Bayesian design method is sketched in algorithm 5.1. Here, d new rows are appended to $\mathbf{X}$ in each iteration ($d = 3$ in our experiments in section 5.5.2).

5.4 Related work and extensions

In this section, we describe work related to ours, focusing on methods that we compare against in section 5.5.2. We also comment on constrained design optimisation within our framework.

Typically, CS reconstruction from incomplete measurements [Candès et al., 2006, Donoho, 2006a] is done by minimising a norm penalty under some sharp observation constraints

$$\hat{\mathbf{u}} = \arg\min_{\mathbf{u}}\{\|\mathbf{B}\mathbf{u}\|_p \text{ s.t. } \mathbf{X}\mathbf{u} = \mathbf{y}\},\ p \in \{1, 2\}. \tag{5.5}$$

Here, $\|\mathbf{s}\|_2 := \sqrt{\mathbf{s}^\top\mathbf{s}}$ denotes the L_2 norm. Maximum sparsity in $\mathbf{s} = \mathbf{B}\mathbf{u}$ is obtained for $p = 0$, yet this L_0 estimation problem is NP hard. If $p = 1$ is chosen instead, the corresponding solution can be found efficiently by solving a linear program. In highly sparse situations, this LP relaxation yields the exact solution to the L_0 problem [Donoho, 2006b]. In our experiments below, we consider several special cases. The simplest CS method (called L_1) is obtained by choosing $p = 1$ and $\mathbf{B} = \mathbf{W}$ (the wavelet transform). It is also known as basis pursuit [Chen et al., 1999]. Classical least squares estimation (called L_2) uses $p = 2$ and $\mathbf{B} = \mathbf{W}$. Since $\mathbf{B}$

Algorithm 5.1 *Sequential Bayesian experimental design*

```
Require: Initial $\mathbf{X}$, $\mathbf{y}$, $\tau_{sp}$, $\tau_{tv}$, $\sigma^2$
  $\boldsymbol{\beta} = \mathbf{0})$, $\boldsymbol{\gamma} = 2[\tau_D^{-2}\mathbf{1}; \tau_W^{-2}\mathbf{1}]$
  Compute initial $\mathbb{Q}$ representation, marginals $(\mathbf{h}, \boldsymbol{\rho})$
  repeat
    $J = \{1, .., q\}$ (for first update)
    repeat
      Compute $\Delta_i = \mathrm{KL}[\mathbb{Q}'_i \,\|\, \mathbb{Q}]$ for all $i \in J$, using $(\mathbf{h}, \boldsymbol{\rho})$.
      EP update at potential $\hat{i} = \arg\max_{i \in J} \Delta_i$.
      Update of $\mathbb{Q}$ representation, marginals $(\mathbf{h}, \boldsymbol{\rho})$.
      Evolve $J$ (shrink to desired size after first iteration).
    until $\Delta_{\hat{i}}$ below threshold
    Find $\mathbf{X}_* \in \mathbb{R}^{d \times n}$: $d$ leading unit norm eigendirections of $\mathbb{V}_{\mathbb{Q}}[\mathbf{u}]$ (Lanczos algorithm).
    Measure image with $\mathbf{X}_* \to \mathbf{y}_* \in \mathbb{R}^d$.
    Append $(\mathbf{X}_*, \mathbf{y}_*)$ to $(\mathbf{X}, \mathbf{y})$.
  until $\mathbf{X}$ has desired size, or $\mathbb{Q}(\mathbf{u})$ has desired entropy
```

is orthonormal, we have $\|\mathbf{Bu}\|_2 = \|\mathbf{u}\|_2$, and $\hat{\mathbf{u}}$ is given as solution of the normal equations: $\hat{\mathbf{u}} = \mathbf{X}^\top(\mathbf{X}\mathbf{X}^\top)^{-1}\mathbf{y}$.

We also consider a method with transform sparsity *and* total variation potential [Candès and Romberg, 2004] (called $\mathrm{L}_1 + \mathrm{TV}$ here):

$$\hat{\mathbf{u}} = \arg\min_{\mathbf{u}} \{\tau_W \|\mathbf{Wu}\|_1 + \tau_D \|\mathbf{Du}\|_1 + (2\sigma^2)^{-1} \|\mathbf{y} - \mathbf{Xu}\|_2^2\}.$$

Note that $\mathrm{L}_1 + \mathrm{TV}$ is the MAP estimator (equation 5.4) for the same model we employ in our Bayesian method. It is also known as the Lasso [Tibshirani, 1996] or the penalised least squares estimators of chapter 2.2.1. L_2 and L_1 (equation 5.5) can be seen as MAP estimators as well, if the noise variance σ^2 is set to zero, so that the likelihood constraints have infinite weight (see section 5.3).

The algorithm of Ji and Carin [2007] is called BCS. It comes with a transform sparsity potential only, so that $\mathbf{s} = \mathbf{Wu}$. BCS employs sparse Bayesian learning [Tipping, 2001] in order to approximate Bayesian inference. This technique is specific to sparse linear models (all $\mathcal{T}_i$ have to be Gaussian scale mixtures, thus even functions), while EP can be applied with little modification to models with skew priors or non-Gaussian skew likelihoods as well [Gerwinn et al., 2008]. We used the following code in our experiments.

$\mathrm{L}_1 + \mathrm{TV}$	`http://www.acm.caltech.edu/l1magic/`
L_1	`http://www.stanford.edu/~mlustig/`
BCS	`http://www.ece.duke.edu/~shji/BCS.html`

5.4.1 Wavelet transformation code

In order to have simple and efficient implementation of the wavelet transforms for tensors, we set up the `FWTN` package. The `FWTN` code includes a standalone implementation of orthonormal wavelet transforms for D-dimensional tensors in L levels. It is generic in the quadrature mirror filter, so any filter (Haar, Daubechies etc.) can be used. Runtime is $\mathcal{O}(n)$ with n being the number of elements of the tensor. The code is written in plain C; a Matlab/Octave mex wrapper as well as a demo is provided. In Matlab, you simply type the following to perform the transformation.

```
qmf = [1,1]/sqrt(2); % Haar Wavelet
L = 3;               % # Levels in the pyramid
W = fwtn(X,L,qmf);   % apply FWTN
```

```
% Daubechies 4 Wavelet
qmf = [1+sqrt(3), 3+sqrt(3), 3-sqrt(3), 1-sqrt(3)]/sqrt(32);
Z = ifwtn(W,L,qmf); % apply inverse transform
```

Code is available from `http://www.kyb.tue.mpg.de/bs/people/hn/fwtn.zip` or the corresponding `mloss.org` project `http://mloss.org/software/view/242/`.

5.4.2 Optimisation of designs under constraints

In our study on optimising image measurements, we assume that filters can be chosen anywhere on the unit sphere. In typical applications of this scenario, additional constraints have to be placed on the rows of $\mathbf{X}$. For example, in magnetic resonance imaging, Fourier coefficients are measured along constrained paths in Fourier space. Or in digital photography, cameras may not be able to realise arbitrary filters $\mathbf{x}_*$ (see chapter 2.6.4).

In many scenarios in practise, the number of candidates $\mathbf{x}_*$ for the next measurement is finite and rather small [Seeger et al., 2007]. In this case, called transductive design, it is easiest to score all candidates and pick the one maximising the information criterion. In one setup in section 5.5.2, we restrict our Bayesian experimental design technique to select among wavelet coefficient filters only. This case is very simple to deal with, since these coefficients feature in the transform sparsity prior potential. If $\mathbf{x}_* = \mathbf{b}_j$ is such a filter, then $\mathbf{x}_*^\top \mathbb{V}_{\mathbb{Q}}[\mathbf{u}]\mathbf{x}_*$ is simply the variance of $\mathbb{Q}(s_j)$, where $\mathcal{T}_j(s_j)$ is the corresponding prior potential. If selective potential updating is used (see section 5.3.3), the variances for all these s_j are maintained at all times, so the optimisation over all wavelet coefficient filters comes almost for free. Obviously, the marginals of *any* other set of linear projections of $\mathbf{u}$ can be kept up-to-date alongside as well, independently of whether they feature in the potentials of the model. Therefore, any extension of the setting considered here, based on a fixed candidate set, where the matrix containing all candidate filters as rows admits a fast matrix-vector product, can be implemented very efficiently.

However, in general the problem of maximising our information criterion, subject to further constraints, is not convex. The function $\mathbf{x}_*^\top \mathbb{V}_{\mathbb{Q}}[\mathbf{u}]\mathbf{x}_*$ is convex in $\mathbf{x}_*$, and the maximisation of a convex function, subject to convex constraints, can be hard. If the constraint set is a ball w.r.t. some Euclidean norm, centred at zero, the optimal $\mathbf{x}_*$ is a (generalised) eigenvector, which is what we use in our setup here. In general, we recommend the simple approach of keeping marginals up-to-date for a finite grid of candidate constraints, then to start some nonlinear optimisation method from the maximiser $\mathbf{x}_*$ on this grid.

5.5 Experiments

In this section, we provide experimental results for different instances of our framework, comparing to CS estimation and approximate Bayesian methods on synthetic data (section 5.5.1), and on the task of measuring natural images (section 5.5.2).

5.5.1 Artificial setups

It is customary in the CS literature to test methods on synthetic data, generated following the "truly sparse and otherwise unstructured" assumptions under which asymptotic CS theorems are proven. We do the same here, explicitly using the "(non-)uniform spikes" [Ji and Carin, 2007], but cover some other heavy-tailed distributions as well. It seems that not many signals of real-world interest are strictly and randomly sparse, so that studies looking at the *robustness* of CS theoretical claims are highly important. In this section, signals are sparse as such, so that $\mathbf{B} = \mathbf{I}$ and $\mathbf{u} = \mathbf{s}$ here. We compare methods described in section 5.1 and section 5.4. It is important to stress that all methods compared here (except for L_2) are based on exactly the same underlying model (equation 5.3) with $\mathbf{B} = \mathbf{I}$, and differences arise only in the nature of computations (approximate Bayesian inference versus maximum a-posteriori estimation), and in whether **X** is sequentially designed (EP, BCS) or chosen at random (L_p reconstruction; we follow CS theory [Candès et al., 2006, Donoho, 2006a] and sample rows of **X** uniformly of unit norm). Results are shown in figure 5.2.

The "sparsity" (or super-Gaussianity) of the signal distributions increases from (5.2a) to (5.2e-f). For Gaussian signals (5.2a), L_2 reconstruction based on random measurements is optimal. While all CS methods and BCS (random and designed) lead to large errors, EP with design matches the L_2 results, thus shows robust behaviour. For Laplacian and Student's t signals (5.2b-c), designed EP outperforms L_2 reconstruction significantly, while even the CS L_1 method still does worse than simple least squares. BCS performs poorly in all three cases with signals not truly sparse, thus is not robust against rather modest violations of the strict CS assumptions. Its non-robustness is also witnessed by large variations across trials.

On the other hand, L_2 performs badly on truly sparse signals. In all cases (5.2d-f), EP with design significantly outperforms all other methods, including designed BCS, with special benefits at rather small numbers of measurements. BCS does better now with truly sparse signals, and is able to outperform L_1.

From the superior performance of EP with design on all signal classes, we conclude that experimental design can sequentially find measurements that are significantly better than random ones, even if signals are truly sparse. Moreover, the superior performance is robust against large deviations away from the underlying model, more so even than classical L_1 or L_2 estimation. The poor performance of BCS [Ji and Carin, 2007] seems to come from their desire for "premature sparsification". During their iterations, many γ_i are clamped to 0 early in the optimisation for efficiency reasons. This does not hurt mean predictions from current observations much, but affects their covariance approximation drastically: most directions not supported by the data at present are somewhat ruled out for further measurements, since the posterior variance along them (which should be large) is shrunk in their method. In contrast, in our EP method, none of the γ_i becomes very tiny with modest m, and our covariance approximation seems good enough to successfully drive experimental design. Without premature sparsification, our scheme is still efficient, since the most relevant potential updates are found actively, and the need to eliminate variables does not arise.

5.5.2 Natural images

In this section, we are concerned with finding linear filters which allow for good reconstruction of natural images from noisy measurements thereof. Natural images exhibit sparsity in a wavelet domain, fulfilling the basic requirement of CS. Theoretical results seem to suggest that measurement filters can be drawn at random, and there have been considerable efforts to develop hardware which can perform such random measurements cost-efficiently [Duarte et al., 2008]. On the other hand, much is known about low-level natural image statistics, and powerful linear measurement transforms have emerged there, such as multi-scale wavelet coefficients, based on which natural image reconstruction should be more precise than for random measurements [Weiss et al., 2007].

The sparsity of images in the wavelet domain is highly structured, there is a clear *ordering* among the coefficients from coarse to fine scales: natural images typically have much more energy in the coarse scale coefficients, and coefficients with very small values are predominantly found in the fine scales. In our experiments, we employ a simple heuristic for linearly measuring images, called *wavelet heuristic* in the sequel: every measurement aims for a single wavelet coefficient, and the sequential ordering of the measurements is deterministic top-down, from coarse to fine scales[7]. This ordering is a pragmatic strategy: if mainly the coarse scale coefficients are far from zero, they should be measured first. Do state-of-the-art CS reconstruction algorithms, based on random linear image measurements, perform better than simple L_2 reconstruction based on the wavelet heuristic? And how does Bayesian sequential design perform

[7] This ordering follows the recursive definition of such transforms: downsampling by factor two (coarse), horizontal differences, vertical differences, diagonal corrections at each stage. Our ordering is coarse → horizontal → vertical → diagonal, descending just as the transform does.

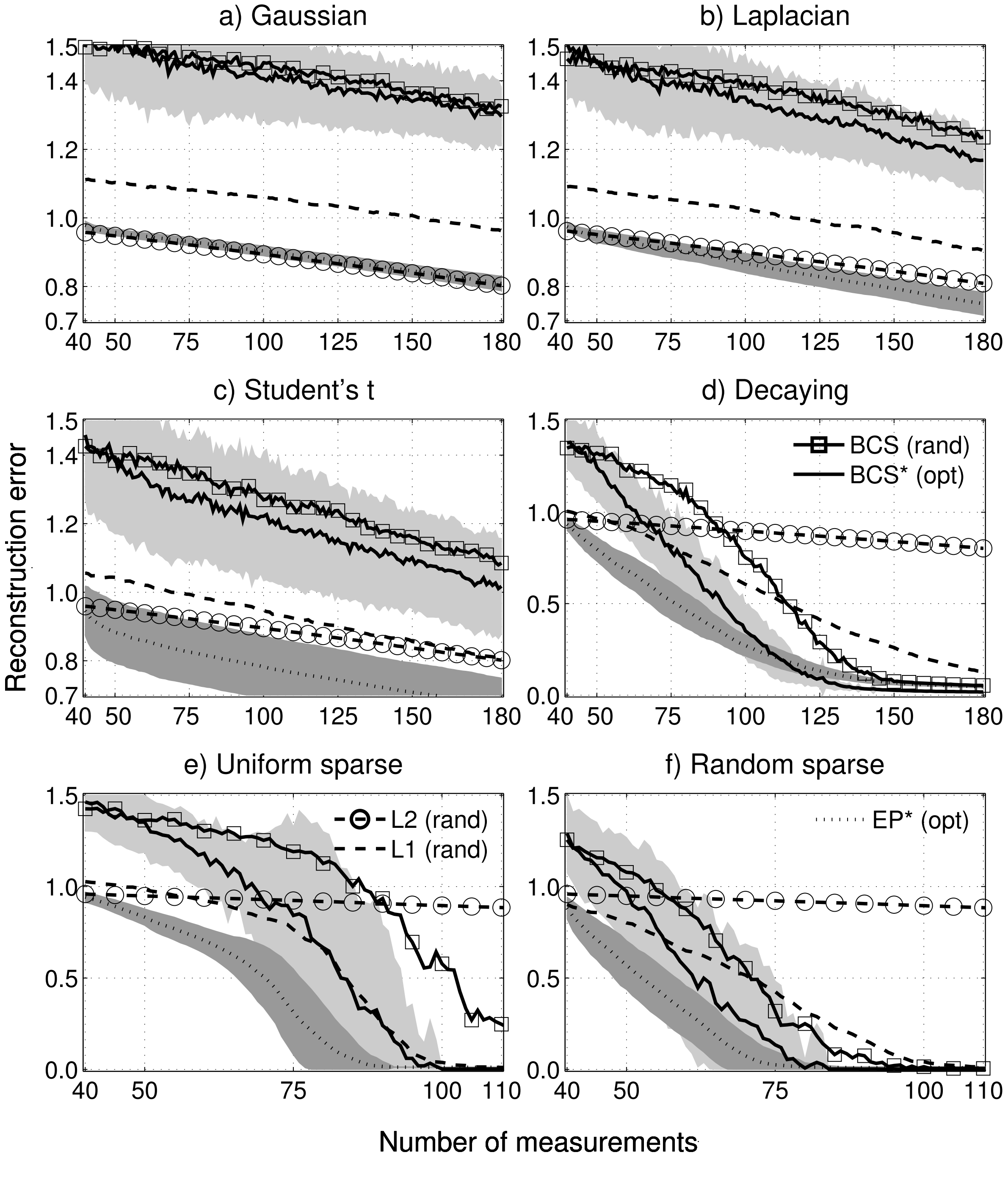


Figure 5.2: Comparison of measurement design on 6 random synthetic signals $\mathbf{u} \in \mathbb{R}^{512}$. Shown are L$_2$*-reconstruction errors (mean±stdard deviation over 100 runs). All methods start with the same random initial* $\mathbf{X}$ *(*$m = 40$*), then "(rand)" add random rows, "(opt)" optimise new rows sequentially. Noise variance* $\sigma^2 = 0.005$*, prior scale* $\tau = 5$*. BCS:* Lp*:* L$_p$ *reconstruction, EP: our method. a-c): i.i.d. zero mean, unit variance Gaussian, Laplacian (equation 5.2), Student's t with* $\nu = 3$*. d):* $\frac{n}{2}$ *of* $u_i = 0$*,* $\frac{n}{4}$ *exponential decay* $1, \ldots, 0$*,* $\frac{n}{4}$ *minus that, randomly permuted. e-f):* 20 $u_i \neq 0$ *at random; (e) uniform spikes,* $u_i \in \{\pm 1\}$*; f): non-uniform spikes,* $u_i \sim \frac{1}{4} + |t|$*,* $t \sim \mathcal{N}(0, 1)$*; as in Ji and Carin [2007]. Distributions in d-f) normalised to unit variance.*

		reconstruction method				
type of design **X**	adapt	L_1	$L_1 + \mathrm{TV}$	L_2	BCS	EP
rand uni	–	abef	abef		a	
rand coloured	–		b			
rand noiselet	–	b	b			
heur wave	–	$=L_2$	d	**a-f**	$=L_2$	d
opt free	+				a	**a-f**
opt wave **panel e) and f)**	+			c		cef

Table 5.1: Experiment summary matrix for figure 5.4

Figure 5.3: Image dataset used for the experimental design benchmark. We benchmarked the algorithms on 75 images frequently used in computer vision research. The bitmaps were obtained from `http://decsai.ugr.es/cvg/dbimagenes/g512.php`.

on this task, if the model described in section 5.1 is used? Furthermore, how strong is the impact of the total variation potential? Note that no prior knowledge about typical ordering or dependence among wavelet coefficients is encoded in this model either.

Recall from section 5.1 that every CS method has to address two problems: reconstruction of the signal $\mathbf{u}$ from measurements $\mathbf{y}$ for a fixed design $\mathbf{X}$, and the choice of the design $\mathbf{X}$. In our experiments, we pair five different reconstruction methods (L_1, $L_1 + \mathrm{TV}$, L_2, BCS, and EP; see section 5.4) with a number of non-adaptive (rand uni, rand coloured, rand noiselet, heur wave) and adaptive (opt free, opt wave) measurement designs. The pairings we explored are summarised in table 5.1. For *rand uni*, entries are drawn uniformly at random: $\mathbf{X}_{ij} \sim \mathcal{N}(0, \frac{1}{n})$. For *rand coloured*, filters are drawn respecting the second order structure of images. Inspired by Wang et al. [2007], we applied a spectral low-pass filter to random Gaussian noise with a power spectrum decaying with f^{-2} [Field, 1987]. For *rand noiselet*, we selected random rows of the noiselet transform [Coifman et al., 2001], as was proposed for CS on images in Candès et al. [2006]. We are grateful to Emmanuel Candès and Justin Romberg for providing us with their noiselet transform code. Finally, *heur wave* is the wavelet heuristic described above. While this heuristic is non-adaptive, in that the ordering is fixed in advance, we also considered the adaptive variant proposed in Dekel [2008] (called *heur Dekel* below). We acknowledge Shai Dekel for sharing code and knowledge with us. The adaptive designs are both sequential, in that new rows $\mathbf{x}_*$ are added to $\mathbf{X}$ one at a time, based on all previous measurements. In *opt free*, the optimisation is done over all unit norm filters $\mathbf{x}_*$, while in *opt wave*, each filter has to correspond to a single wavelet coefficient. Note that *opt wave* is another adaptive alternative to the wavelet heuristic. The database for our study is a set of 75 natural grey-scale images frequently used in computer vision research (figure 5.3), which were scaled to 64 × 64 pixels. Results are given in the panels of figure 5.4 (legend entries have the form "reconstruction method (type of design)").

In the main panel a), we consider natural pairings: our Bayesian EP method, as well as BCS, with unconstrained experimental design (opt free), and current CS reconstruction meth-

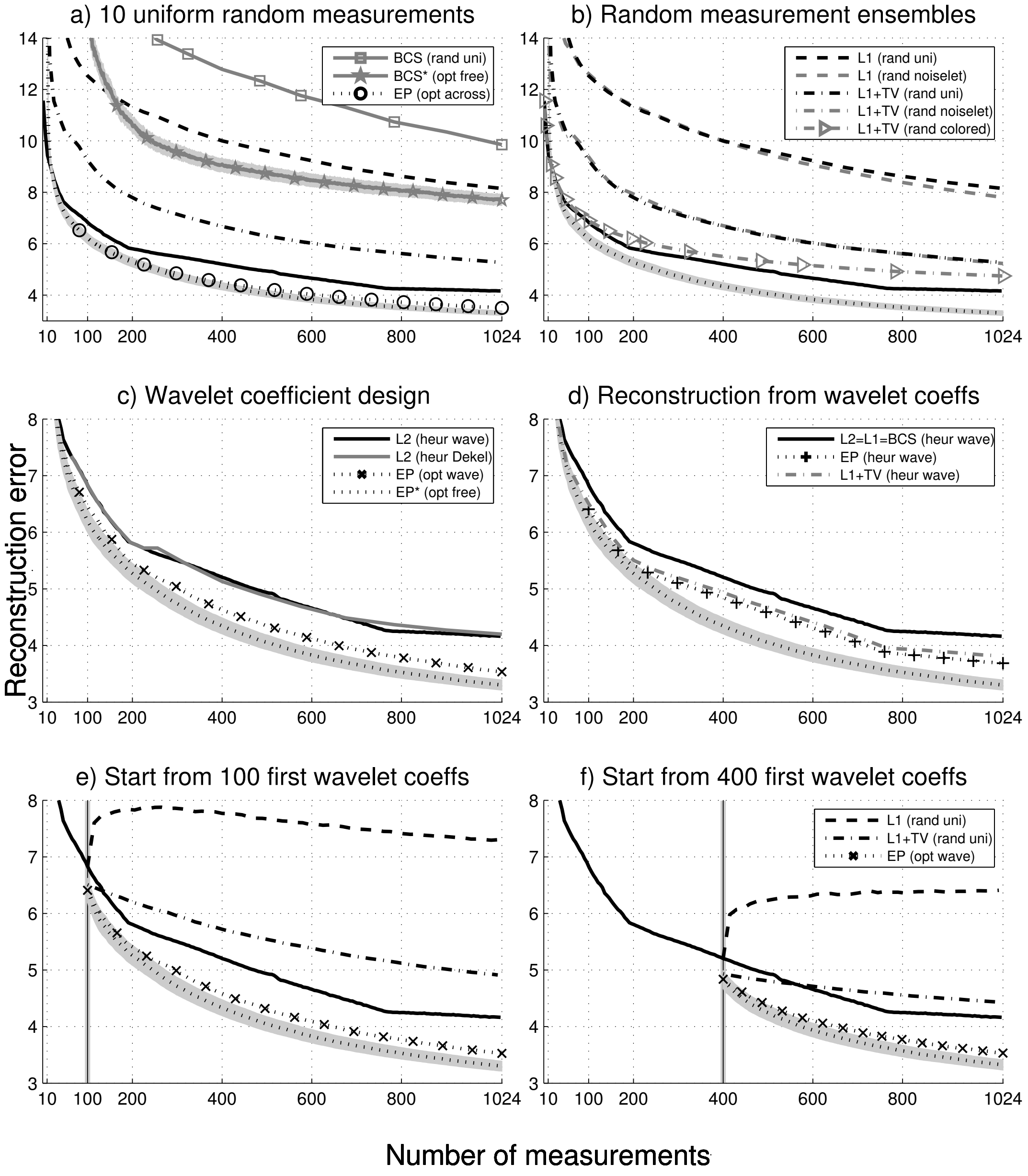


Figure 5.4: Comparative results for the experimental design benchmark.
Experiments for measuring natural images of size $64 \times 64 = 4096$ *pixels depicted in figure 5.3. Shown is* L_2*-reconstruction error averaged over 75 grey-scale images (±standard error of the mean for "$*$"). Noise level* $\sigma^2 = 0.005$*. BCS:* Lp*:* L_p *reconstruction* $p \in \{1, 2\}$, $\mathrm{L}_1 + \mathrm{TV}$*: Lasso with TV/wavelet penalties, EP: our method. True* σ^2 *supplied,* τ *parameters chosen optimally for each method individually:* $\tau_W = \tau_D = 0.075$ $(\mathrm{L}_1 + \mathrm{TV})$*,* $\tau_W = 0.075$*,* $\tau_D = 0.5$ *(EP). New rows* $\mathbf{x}_*$ *of* $\mathbf{X}$ *random unit norm (rand), actively designed (opt), according to wavelet heuristic (heur wave).*
a) Start from $m = 10$ *with* $\mathbf{X}$ *random uniform. b) Comparison for* $\mathbf{X}$ *drawn from different measurement ensembles. c) Optimisation restricted to wavelet coefficients. d) Different reconstruction methods based on same measurements (heur wave). e,f) Start from* $m = 100, 400$ *with* $\mathbf{X}$ *according to wavelet heuristic. See table 5.1 for a complete list.*

ods (L_1, $\mathrm{L}_1 + \mathrm{TV}$) with randomly drawn measurement filters (rand uni). The wavelet heuristic is paired with least squares reconstruction (L_2). Note that *EP(opt free)* and L_2*(heur wave)* feature in all panels for reference. All methods in a) are started from ten initial filters drawn according to *rand uni*, except for *BCS(opt free)*, which required 100 initial filters (*rand uni*) to attain a decent performance. The L_2 wavelet heuristic clearly outperforms all CS methods based on random designs. Among the latter, $\mathrm{L}_1 + \mathrm{TV}$ does substantially better than L_1 or BCS, indicating the importance of the total variation prior potential. This is also witnessed in the scale parameters employed for the two potentials in EP: $\tau_W = 0.075$, $\tau_D = 0.5$. The total variation potential is much stronger. In fact, the results of EP with $\tau_W = 0$, $\tau_D = 0.5$ are only insignificantly worse. Note that the BCS code supplied with Ji and Carin [2007] allows for a transform sparsity potential only. Moreover, our method *EP(opt free)* outperforms the wavelet heuristic, by selecting filters which are more informative than wavelet coefficients. Since *EP(opt free)* adjusts the design $\mathbf{X}$ specifically for each underlying image, it is natural to ask whether such designs are transferable to other images as well. In the setup *EP(opt across)*, we reconstructed each image $\mathbf{u}$ using five measurement designs $\mathbf{X}$ adapted to *different* images (randomly chosen). The average reconstruction error is shown in a): as expected, it is slightly worse than for *EP(opt free)*, yet still substantially better than the L_2 wavelet heuristic. Therefore, the filters found by *EP(opt free)* turn out to be transferable to other images, opening up the possibility to adapt such designs *a priori*. Finally, the poor performance of BCS, compared to the simpler L_1 or $\mathrm{L}_1 + \mathrm{TV}$, is remarkable.

In panel b), we consider other ensembles beyond *rand uni*, which the designs $\mathbf{X}$ are drawn from. The random noiselet ensemble *rand noiselet* proposed for CS in Candès et al. [2006] has the theoretical advantage of being maximally incoherent with the Haar wavelet basis. Moreover, $\mathbf{X}$ does not have to be stored explicitly in this case, and MVMs with $\mathbf{X}$ or $\mathbf{X}^\top$ can be computed very efficiently. There is no significant difference between *rand uni* and *rand noiselet* for $\mathrm{L}_1 + \mathrm{TV}$. While the noiselet measurements lead to a more compact algorithm, they do not result in better reconstructions. The coloured ensemble *rand coloured* results in filters more closely aligned with the signal energy. They lead to significant improvements over the uninformed ensembles, yet are again outperformed by the L_2 wavelet heuristic.

In panel c), we compare adaptive alternatives to the wavelet heuristic. The heuristic proposed in Dekel [2008] does not improve upon L_2*(heur wave)* in our experiments. However, our EP method significantly outperforms the heuristic, even when constrained to measure wavelet coefficients only (see section 5.4.2). The advantage may be due to EP choosing a better ordering of the coefficients, but also due to improved reconstruction (see also panel d). While *EP(opt free)* still outperforms the constrained variant *EP(opt wave)*, we see that the design optimisation of our method is successful under structural constraints on the filters as well.

In panel d), we try to separate between reconstruction performance and the choice of measurement design. All methods shown there use the same wavelet heuristic design (except for *EP(opt free)*, added for reference). First of all, L_2, L_1, and BCS provably give exactly the same reconstruction, if $\mathbf{X}$ is a part of $\mathbf{W}$. $\mathrm{L}_1 + \mathrm{TV}$ and EP can do significantly better based on these measurements, while there is no significant difference between them. It is also interesting to compare *EP(heur wave)* here with *EP(opt wave)* in panel c). The latter does slightly better, although the major part of the improvement over L_2*(heur wave)* is due to EP being a better reconstruction method.

Intrigued by the fact that the wavelet heuristic with simple L_2 reconstruction outperformed *all* estimators based on random designs, we analysed their performance after giving them a warm-start, by supplying them with the first 100 and first 400 wavelet heuristic measurements. The results are shown in **panel e) and f)** respectively. In this setting, BCS with projection optimisation performed much worse than all other methods, the results are omitted to facilitate the comparison among the others. $\mathrm{L}_1 + \mathrm{TV}$ profits from the warm-start to some extent, although its final performance (continuing with *rand uni*) is worse than the L_2 wavelet heuristic. Both *EP(opt free)* and *EP(opt wave)* improve upon L_2*(heur wave)* from the moment they are allowed to choose filters by themselves, so the warm-start is in fact suboptimal for them. The deterioration of L_1 is rather striking, given that additional measurements provide novel information about the true $\mathbf{u}$. The failure is analytically explained in appendix G.1.

From these results we conclude, much as Weiss et al. [2007] argued on mostly theoretical grounds, that if natural images are to be measured successively by unit norm, but otherwise unconstrained linear filters, drawing these filters at random leads to significantly worse reconstructions than standard wavelet coefficient filters top-down. Moreover, the wavelet heuristic can be improved upon by adapting filters with our Bayesian experimental design technique. To put our findings into perspective, we note that the L_2 wavelet heuristic is vastly faster to compute[8] than all other methods considered here. Another finding is that the total variation potential seems to be more useful for natural images than the transform sparsity term. Our Bayesian design optimisation method, based on EP, can be used under structural constraints, and the designs can successfully be transferred to measure other images as well. CS theorems are mathematically intriguing, and there are certainly applications that benefit from these results, but linear image measurement is probably not among them.

Possible reasons for the failure of BCS on signals that are not truly sparse, were given in section 5.5.1. Premature sparsification, in light of not strictly sparse signals, leads to poor results even with random $\mathbf{X}$. Their covariance estimates seem too poor to steer sequential design in a useful direction.

5.6 Discussion

We have shown how to address the CS problem with Bayesian experimental design, where designs are optimised to rapidly decrease uncertainty, rather than being chosen at random. In a study about linearly measuring natural images, we show that CS reconstruction methods based on randomly drawn filters are outperformed significantly by standard least squares reconstruction measuring wavelet coefficients in a fixed ordering from coarse to fine scales. Our findings suggest that the impact of CS theoretical results to natural image applications should be reconsidered. We also show that our Bayesian sequential design method, starting from a model with little domain knowledge built in, is able to find filters with significantly better reconstruction properties than top-down wavelet coefficients. Our findings indicate that efficient Bayesian experimental design techniques such as ours should be highly promising for CS applications in general.

Our best explanation for the differences between theory and what is found in natural image applications, is based on the explicit *worst-case* character of the theorems: while the signal is assumed to be sparse in some transform domain, no assumptions are made about where the non-zeros lie. Moreover, the statements are usually of the *minimax* type, bounding the performance or success probability under the worst possible placing of the non-zero set. It is reassuring that random measurements and simple convex estimation methods are sufficient to give useful results within broad regimes of such a pessimistic setting. The impact in applications, where high standards of security have to be met, or where adversarial signal constructions have to be detected, may be substantial. However, in practical statistics, worst-case results are often not transferable to "cases of practical interest". While it is easy to see that experimental design can fail badly in the worst case, a proper implementation often leads to significant cost reductions for non-adversarial tasks, whose properties can be modelled well. In minimax techniques, available prior knowledge can often be ignored, because the worst case may just as well be very unexpected. Moreover, making decisions about future sampling based on data observed so far, is usually not useful, because the "benign" assumptions underlying these techniques are violated in the worst case. It is therefore not reasonable to conclude from minimax results, or from results assuming the absence of any structure except for sparsity, that methods which perform close to optimal in these cases, set the standard in practise as well. In fact, while

[8] EP sequential design is still very efficient. A typical run on one image took 53 minutes (on 64bit 2.33GHz AMD), for $n = 4'096$ and $q = 12'160$ potentials: $16'785$ initial EP updates, then 308 increments of $\mathbf{X}$ by 3 rows each, with on average only 8.8 potential updates needed to regain EP convergence (up to 85 updates after some increments).

minimax CS theory requires **X** and **B** to be as "incoherent" w.r.t. each other as possible [Candès et al., 2006], and some methods strive for maximally incoherent designs [Elad, 2007, Candès et al., 2006], on natural images, these methods are significantly outperformed by using wavelet coefficients in a certain ordering. The latter filters are rows of **B**, therefore *maximally coherent* with the sparsifying transform. If wavelet coefficients were sparse at random for the ensemble of natural images, incoherence would indeed be an important property of a measurement design. Since the sparsity of images is structured in a stable way, the completely coherent wavelet heuristic performs much better than worst-case optimal incoherent designs.

Our experience with the method of Ji and Carin [2007], which we compare against in our study, raises another interesting question. Several signal processing and machine learning methods try to detect sparsity early on for computational efficiency. Sparse Bayesian learning [Tipping, 2001] is more aggressive in this respect than our EP method here. Early sparsification seems to not hurt mean prediction performance much. However, our experiences indicate that it is the *covariance* (or uncertainty) estimates that can be badly hurt by sparsity-by-elimination, and that in contexts such as experimental design, where covariances are more important than predictive means, they should be avoided. The challenge is to develop methods that run efficiently *without* eliminating many variables early on, and our selective potential updating method for EP is a step in that direction.

Chapter 6

Magnetic Resonance Imaging Sequence Optimisation

Magnetic resonance imaging (MRI) is one of the most widely used medical imaging modalities and offers excellent soft tissue resolution without exposing the patient to unhealthy radiation. Most of the research effort today aims at increasing the spatial and temporal resolution by optimising the scanner hardware and the MR measurement sequence. Another recent approach to speed up MRI *undersamples* the signal and uses sparse estimation algorithms for faithful image reconstruction from incomplete measurements [Lustig et al., 2007]. Sparse estimation algorithms exploit stable low-level statistical properties that strongly constrain the class of proper images: unlike random noise, natural and medical images are defined by edges and smooth areas. While the majority of clinically used sequences have a reconstruction cost of a single fast Fourier transform (FFT), iterative sparse reconstruction techniques require several of these: in a nutshell, sparse reconstruction algorithms trade faster measurements against higher computational load afterwards.

A different, but related and more difficult problem is to design and improve the undersampling sequences, producing the data for subsequent sparse reconstruction, themselves. We describe a Bayesian method, that maintains a posterior distribution over images that quantifies the uncertainty attached to the image; we view image reconstruction as an inference problem from incomplete noisy information starting from a non-Gaussian prior distribution that captures low-level spectral and local natural image statistics. The posterior is used to judge the quality of the current sequence and the expected improvement after alteration: we sequentially modify the sequence to decrease uncertainty in regions or along directions of interest. Importantly, we do not need to run MRI experiments to score the possible modifications – this is done by our probabilistic computational model.

Based on theoretical results, it has been proposed to design sequences by randomising aspects thereof [Lustig et al., 2007]. Beyond being hard to achieve on a scanner, our results indicate that randomised measurements do not work well for real MR images. Similar negative findings for a variety of natural images were also given in chapter 5. Our algorithm enables efficient Bayesian inference computations for MR images of realistic resolution. The inference problem is reduced to numerical mathematics primitives, and further to matrix-vector multiplications (MVM) with large, structured matrices, which are computed by efficient signal processing code. Based on raw data from a 3T MR scanner, we apply our sequence optimisation approach to the design of Cartesian and spiral trajectories, achieving a scan time reduction of a factor larger than two in either case, compared to full sampling. We find that we can indeed improve MRI sequences through the optimisation of Bayesian design scores. Most notably, the improvement transfers to unseen images, which allows to decouple the sequence optimisation and the actual usage of the sequence. Our framework is generic and can be applied to arbitrary trajectory classes, to multi-slice design optimisation [Seeger, 2010b], and to designs with multiple receiver coils.

The general algorithmical idea for approximate inference and experimental design is based on a conference paper [Seeger, Nickisch, Pohmann, and Schölkopf, 2009]; a longer journal paper [Seeger, Nickisch, Pohmann, and Schölkopf, 2010] contributes thorough validation experiments and more MRI material to the chapter.

In section 6.1, we start by introducing the problem of speeding up the MRI acquisition process and some recent efforts exploiting redundancies in the underlying image. We then review basic facts about the MRI measurement process and abstractly introduce the Bayesian design methodology to optimise the measurement process. Later, in section 6.2, we instantiate the probabilistic model using a Gaussian likelihood and a sparse image prior followed by a discussion of point spread functions in linear and nonlinear imaging systems. The inference algorithm is described in section 6.3 starting from a highlevel overview down to a detailed description and some interesting insights. Finally, section 6.4 provides empirical results for Cartesian and spiral measurement trajectories validating our approach to sequence optimisation. Conclusion and perspectives are given in section 6.5.

6.1 Introduction

Magnetic resonance imaging (MRI) [Lauterbur, 1973, Garroway et al., 1974], as a key diagnostic technique in healthcare nowadays, is also of central importance to experimental research of the brain. Without applying any harmful ionising radiation, this technique stands out by its amazing versatility: by combining different types of radio frequency irradiation and rapidly switched spatially varying magnetic fields (called *gradient*s) superimposing the homogeneous main field, a large variety of different parameters can be recorded, ranging from basic anatomy to imaging blood flow, brain function or metabolite distribution. For this large spectrum of applications, a huge number of *sequences* has been developed that describe the temporal flow of the measurement, ranging from a relatively low number of multi-purpose techniques like FLASH [Haase et al., 1986], RARE [Hennig et al., 1986], or EPI [Mansfield, 1977], to specialised methods for visualising bones UTE [Robson et al., 2003], SWIFT [Idiyatullin et al., 2006] or perfusion CASL [Williams et al., 1992]. To select the optimum sequence for a given problem, and to tune its parameters, is a difficult task even for experts, and even more challenging is the design of new, customised sequences to address a particular question, making sequence development an entire field of research [Bernstein et al., 2004]. The main drawbacks of MRI are high initial and running costs, since a very strong homogeneous magnetic field has to be maintained, moreover long scanning times due to weak signals and limits to gradient amplitude.

With this in mind, by far the majority of scientific work on improving MRI is motivated by obtaining diagnostically useful images in less time. Beyond reduced costs, faster imaging also leads to higher temporal resolution in dynamic sequences for functional MRI (fMRI), less annoyance to patients in cardiac examinations or angiography, and fewer artifacts due to patient motion. One way of dealing with the need for rapid scanning are alternative encoding strategies, making use of multiple receiver coils [Sodickson and Manning, 1997, Pruessmann et al., 1999, Griswold et al., 2002] in order to parallelise the measurement process to some degree.

6.1.1 Compressed sensing

While parallel MRI exploits redundancies between several receiver channels, imaging speed can also be increased by taking advantage of redundancies in the signal itself, which allows to reconstruct the image from only a part of k-space in the first place. In MRI, the term k-space denotes the spatial frequency domain or Fourier representation of the image. For example, k-space measurements of real-valued signals show approximately Hermitian symmetry, which is exploited in partial Fourier acquisition techniques [McGibney et al., 1993]. Far beyond these simple symmetries, images form a statistically tightly constrained signal class. Fast, efficient digital image and video compression techniques are routinely used today, and the principles underlying them hold much promise for undersampled high resolution MRI reconstruction [Weaver et al., 1991, Marseille et al., 1996, Wajer, 2001, Lustig et al., 2007], if this process is understood in terms of nonlinear statistical estimation.

These ideas are known as compressed sensing [Candès et al., 2006, Donoho, 2006a] or sparse reconstruction, since they exploit the statistical sparsity of images, a robust low-level characteristic, which leads to nonlinear, yet conservative and well-characterised interpolation behaviour [Weaver et al., 1991]. Compressed sensing is increasingly used for MRI problems, such as dynamic [Gamper et al., 2008] and spectroscopic imaging [Hu et al., 2008], as well as for spiral [Santos et al., 2006] and radial undersampling [Ye et al., 2007, Block et al., 2007]. Typically, scan time reductions by a factor of two or more can be achieved without losses in spatial resolution or sensitivity. Sparse statistics of images or image series originate from the structure of their pixel representations; an important instance is spatial or temporal redundancy, which has been used to speed up MRI acquisition [Korosec et al., 1996, Madore et al., 1999, Tsao et al., 2003, Mistretta et al., 2006].

Two problems arise in practical applications of compressed sensing: how to reconstruct an image from a fixed undersampling design, and how to choose the design in the first place. While a large amount of work was done for the former, we are not aware of much progress for the latter. Although there is substantial prior work on k-space optimisation [Greiser and von Kienlin, 2003, von Kienlin and Mejia, 1991, Spielman et al., 1995], this has been done for linear reconstruction (section 6.2.1), neglecting image sparsity (section 6.2.2). As we demonstrate here, it pays off to match the k-space trajectory to the sparse reconstruction technique. Established concepts such as the point spread function (section 6.2.3), tailored to linear reconstruction, do not capture the inherent dependence of sparse (nonlinear) estimation algorithms on the acquired signal. The latter cannot improve upon the Nyquist limit uniformly, but only for statistically sparse signals, and successful nonlinear k-space optimisation has to take this dependence into account. We phrase k-space optimisation as a problem of experimental design, and propose an algorithm based on Bayesian inference, where statistical sparsity characteristics of images are incorporated by way of a prior distribution. The application of this procedure to high resolution MR images becomes feasible only with the scalable inference algorithm of chapter 3.

Properties of measurement designs for nonlinear sparse reconstruction have been evaluated empirically in Marseille et al. [1996] for Cartesian trajectories, and in [Wajer, 2001, section 6] for radial and spiral trajectories. They focus on non-convex image reconstruction and search for good designs by undirected random exploration, which is unlikely to cover the design space properly. In contrast, we employ the full Bayesian posterior in order to direct our search in a powerful and easily configurable manner. Before we sketch our Bayesian approach to k-space optimisation, we will introduce some MRI terminology and background.

6.1.2 MRI measurement process

An MR scanner acquires Fourier coefficients $Y(\mathbf{k})$ at spatial frequencies $\mathbf{k}$ (the 2d Fourier domain is called k-space) of the proton density $U(\mathbf{r})$ of an underlying object along smooth trajectories $\mathbf{k}(t)$ determined by magnetic field gradients $\mathbf{g}(t)$ as summarised in figure 6.1. The gradient control flow $\mathbf{g}(t)$ in combination with other scanner parameters is called sequence. Its cost is dominated by how long it takes to obtain a complete image, depending on the number of trajectories and their shapes. Gradient amplitude and slew rate constraints due to hardware enforce smooth trajectories.

- In Cartesian sampling, trajectories are parallel equispaced lines in k-space, so the FFT can be used to switch between $Y(\mathbf{k})$ and $U(\mathbf{r})$.
- Spiral sampling offers a better coverage of k-space for given gradient power, leading to faster acquisition. It is often used for dynamic studies, such as cardiac imaging and fMRI. However, since k-space is non-equispacedly covered, we cannot use the FFT algorithm to switch between $Y(\mathbf{k})$ and $U(\mathbf{r})$.

Since the Fourier transformation is a linear operation, the measured data $\mathbf{y}$ is – except for noise – a linear function (depending on the trajectory $\mathbf{k}(t)$) of the underlying object $\mathbf{u}$. Formally,

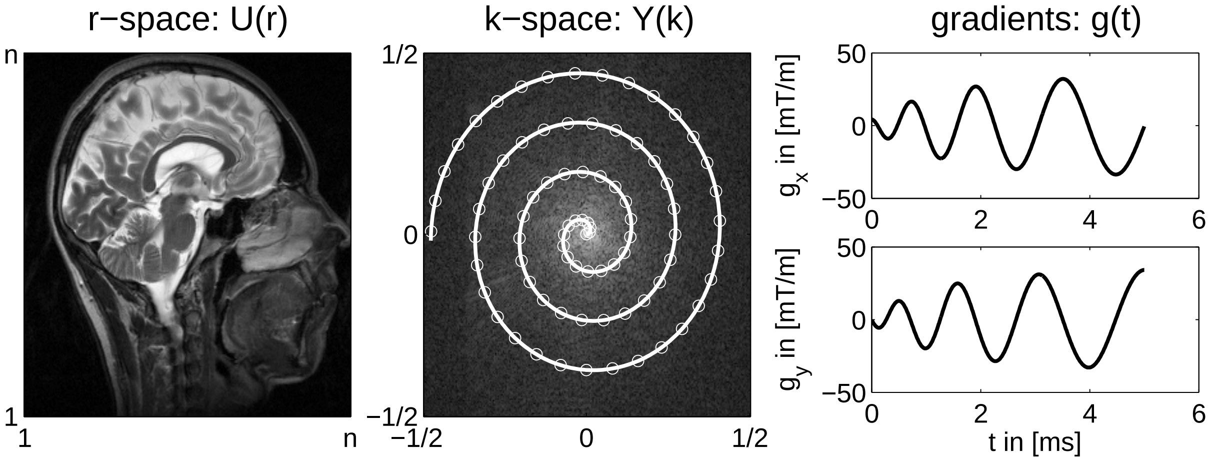


Figure 6.1: MRI signal acquisition
Left: the (proton density of the) underlying object $U(\mathbf{r})$ in 2D pixel space (indexed by $\mathbf{r}$). Middle: the Fourier representation of $U(\mathbf{r})$ in Fourier space is called k-space representation of the signal $Y(\mathbf{k})$. An MR scanner measures along smooth trajectories in k-space (white line). Right: trajectories are obtained by means of spatial magnetic field gradients varying over time. Both Fourier locations $\mathbf{k}$ and spatial locations $\mathbf{r}$ are seen as $\in \mathbb{R}^2$ or $\in \mathbb{C}$.

a trajectory $\mathbf{k}(t)$ leads to data $\mathbf{y} = \mathbf{X_k u}$, where $\mathbf{X_k} = [e^{-\mathrm{i}2\pi \mathbf{r}_j^\top \mathbf{k}(t_\ell)}]_{\ell j}$ is a Fourier matrix. We use gridding interpolation[1] with a Kaiser-Bessel kernel [Bernstein et al., 2004, chapter 13.2] to approximate an MVM with $\mathbf{X_k}$, which would be too expensive otherwise. The matrix $\mathbf{X_k}$ is approximated by **CFD**, where **C** is a banded matrix, **F** is the orthonormal equispaced Fourier matrix and **D** is diagonal allowing for fast multiplications. As for other reconstruction methods, much of our running time is spent in the gridding (MVMs with $\mathbf{X_k}$ and $\mathbf{X_k^H}$).

In theory, the true proton density image $\mathbf{u}_{\text{true}}$ is real-valued; in reality, due to resonance frequency offsets, magnetic field inhomogeneities, and eddy currents [Bernstein et al., 2004, chapter 13.4], the reconstruction contains a phase $\boldsymbol{\varphi}(\mathbf{r})$. It is common practise to discard $\boldsymbol{\varphi}$ after reconstruction leaving the absolute value $|\mathbf{u}_{\text{true}}|$ unchanged. Short of modelling a complex-valued $\mathbf{u}$, we correct for low-frequency phase contributions by a cheap pre-measurement. We sample the centre of k-space on a $p \times p$ Cartesian grid, obtaining a low-resolution reconstruction by FFT, whose phase $\tilde{\boldsymbol{\varphi}}$ we use to correct the raw data. We tried $p \in \{16, 32, 64\}$ (larger p means better correction), results below are for $p = 32$ only. While reconstruction errors generally decrease somewhat with larger p, the relative differences between all settings below are insensitive to p. From the corrected raw data, we simulate all further non-Cartesian measurements under different sequences using gridding interpolation.

With the MR terminology in place, we can now look at our approach to optimise the sequence $\mathbf{k}$ and the measurement design $\mathbf{X_k}$. We write $\mathbf{X}$ for short if $\mathbf{k}$ is clear from the context.

6.1.3 Bayesian k-space optimisation

Within a class of measurement designs $\mathbf{X}$ of equal acquisition cost, which of them leads to the most successful sparse reconstruction of MR images $\mathbf{u}$? While this question has been addressed satisfactorily for linear reconstruction, by the concept of point spread functions, we are not aware of a theory for the nonlinear sparse counterpart. Properties of nonlinear reconstruction are fundamentally signal-dependent, and to our knowledge, no theory at present captures the signal class of high-resolution MR images properly.

Optimising a measurement design $\mathbf{X}$ involves decisions from imperfect information with a quickly growing number of options to choose from. The basic rationale in the following is

[1] Nonequispaced fast Fourier transform (NFFT): `http://www-user.tu-chemnitz.de/~potts/nfft/`

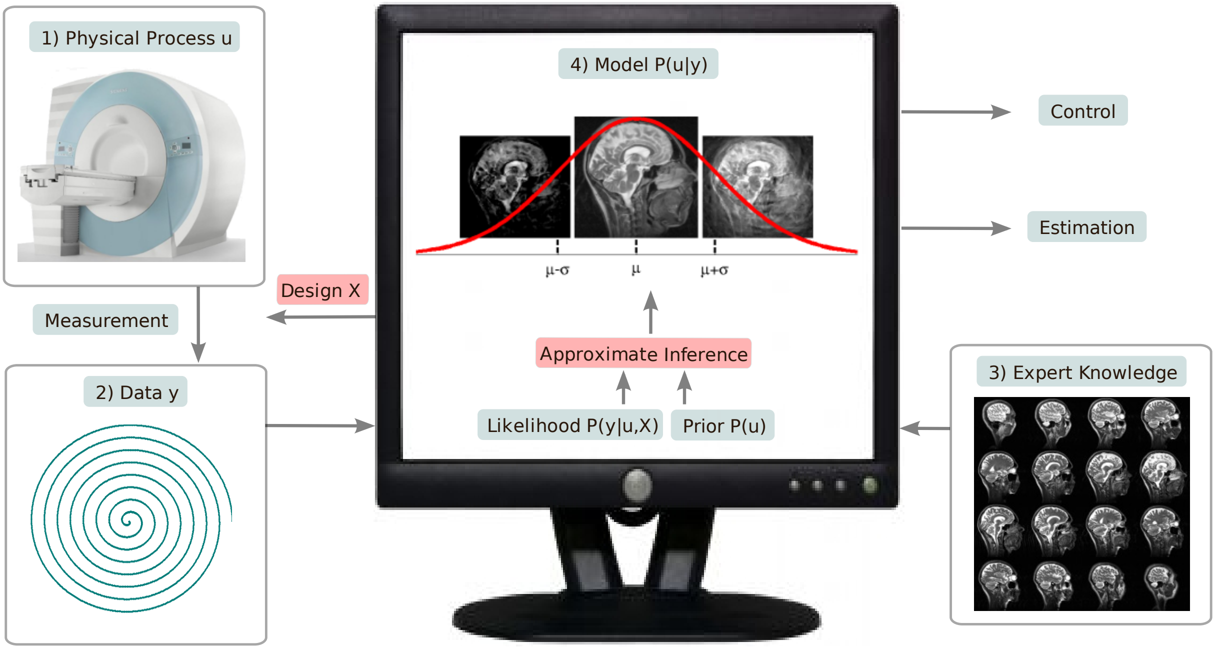


Figure 6.2: Application of experimental design to MRI
Image acquisition using an MR scanner, either by a medical doctor for diagnostic purposes or a researcher in a patient study for scientific reasons, is an interactive process. The scanner 1) measures Fourier coefficients $\mathbf{y}$ of the proton density $\mathbf{u}$ of the tissue under investigation 2), which can be formalised by a likelihood function $\mathbb{P}(\mathbf{y}|\mathbf{u})$. In addition to the data $\mathbf{y}$, one can use prior knowledge 3) given by a distribution $\mathbb{P}(\mathbf{u})$ as an auxiliary input. The internal representation of uncertainty about the image 4) in terms of a posterior distribution $\mathbb{P}(\mathbf{u}|\mathbf{y})$ can be used to derive decisions ranging from refining the image (change the design) or a diagnosis.

to trade expensive computations on computers against human time in decision making under uncertainty.

We develop a variant of Bayesian sequential experimental design (or Bayesian active learning) in this section, in order to optimise k-space sampling automatically from data, specifically for subsequent sparse reconstruction. As illustrated in figure 6.2, the key idea is to monitor the posterior distribution $\mathbb{P}(\mathbf{u}|\mathbf{y})$, the Bayesian representation of remaining uncertainty in the image reconstruction, as the design $\mathbf{X}$ is sequentially extended. Sampling trajectories in most MRI sequences are composed of smooth elements, such as spiral or radial interleaves, or Cartesian phase encodes. Our design algorithm sequentially operates on a candidate set $\mathcal{C} = \{\mathbf{X}_*\}$ of such elements, and appends in each round the element $\mathbf{X}_*$, which leads to the largest expected reduction in posterior uncertainty to the design $\mathbf{X}$ as outlined in algorithm 6.1.

The selection criterion or design score we employ is the *information gain* $IG(\mathbf{X}_*; \mathbb{P}(\mathbf{u}|\mathbf{y}))$ (see chapter 2.26), quantifying the amount of reduction in posterior entropy due to the measurement of an additional phase encode $\mathbf{X}_*$. More precisely, it quantifies the difference in un-

Algorithm 6.1 *Bayesian design optimisation algorithm*

```
Require: Candidate set $\mathcal{C}$ of elements (interleaves, phase encodes). Initial design $\mathbf{X}$, measurement $\mathbf{y}$, corresponding posterior $\mathbb{P}(\mathbf{u}|\mathbf{y})$.
  repeat
    (1) Compute score values $IG(\mathbf{X}_*; \mathbb{P}(\mathbf{u}|\mathbf{y}))$ for all candidate elements $\mathbf{X}_* \in \mathcal{C}$.
    (2) Append winning candidate $\mathbf{X}_*$ to $\mathbf{X}$, and remove it from $\mathcal{C}$.
    (3) Acquire measurement $\mathbf{y}_*$ corresponding to $\mathbf{X}_*$, append it to $\mathbf{y}$.
    (4) Recompute novel posterior $\mathbb{P}(\mathbf{u}|\mathbf{y})$.
  until $\mathbf{X}$ has desired size and $\mathbf{u}$ desired quality
```

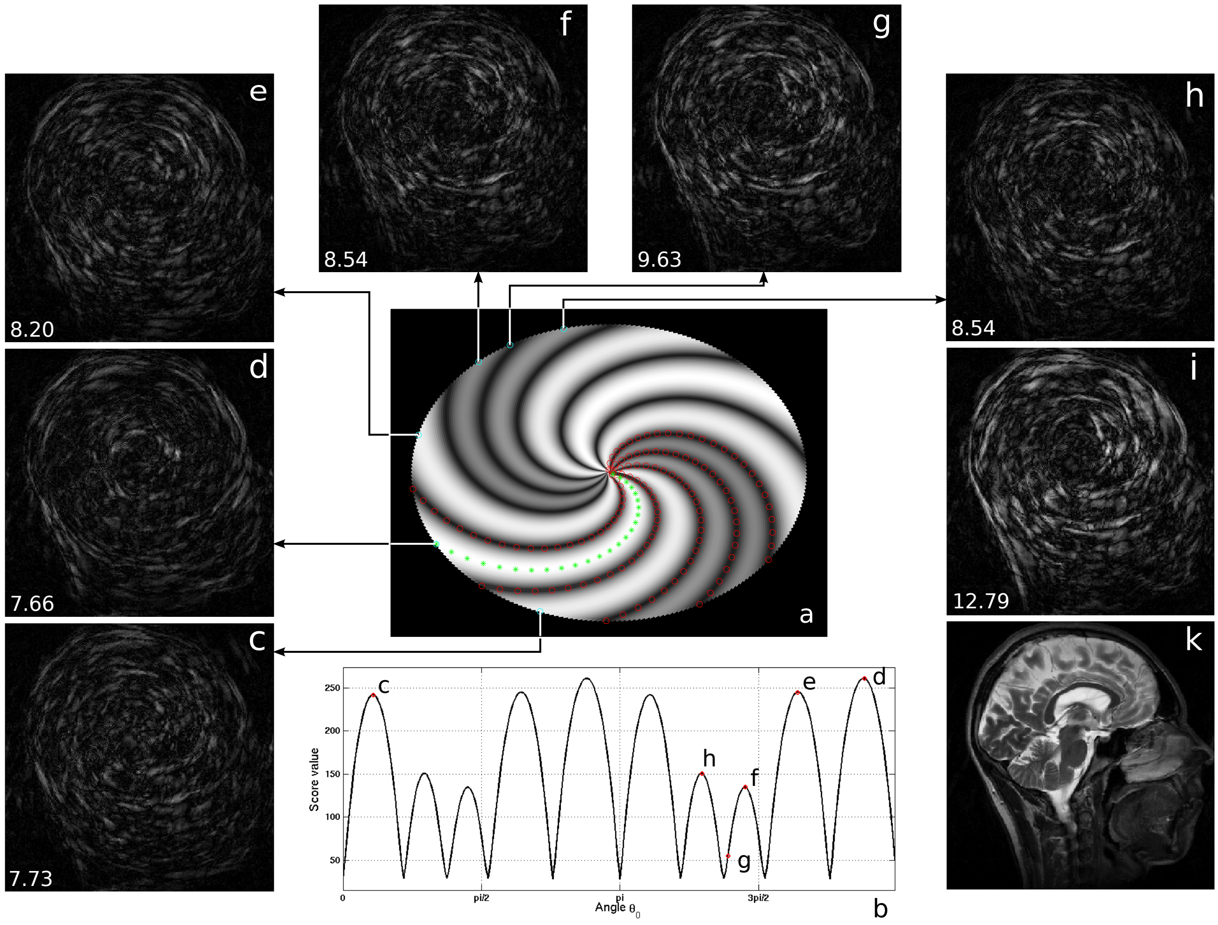


Figure 6.3: Bayesian experimental design on sagittal head scan data for spiral sequences. Five spiral interleaves of the ground truth image (panel a, red dots) have already been measured. The current MAP reconstruction (from the 5 *interleaves in* $\mathbf{X}, \mathbf{y}$*) with respect to the ground truth (panel k) is shown in panel i) along with the reconstruction error.*
The score values $IG(\mathbf{X}_*; \mathbb{Q}(\mathbf{u}|\mathbf{y}))$ *for our* 256 *candidate spirals with outgoing angle* $\theta_0 \in 2\pi \cdot [0..255]/256$ *are visualised by panels a) and b). Panels c–h) show MAP reconstructions from different design extensions* $\mathbf{X} \cup \mathbf{X}_*$*, i.e.* 6 *interleaves (panel a, cyan dots). Shown are residuals* $|\mathbf{u}_* - \mathbf{u}_{true}|$ *for reconstructions* $\mathbf{u}_*$*,* L_2 *error lower left. Top scorer (panel a, green stars) in panel d) gives best reconstruction after extension, due to most information gained. Nontrivial score curve witnesses signal dependence of design optimisation problem.*

certainty between the present state of knowledge $\mathbb{P}(\mathbf{u}|\mathbf{y})$ and the refined state $\mathbb{P}(\mathbf{u}|\mathbf{y}, \mathbf{y}_*)$ after a novel measurement $\mathbf{y}_*$ at $\mathbf{X}_*$. A natural measure for the amount of uncertainty in a distribution $\mathbb{P}(\mathbf{u})$ is the differential entropy $\mathcal{H}[\mathbb{P}(\mathbf{u})] = -\int \mathbb{P}(\mathbf{u}) \log \mathbb{P}(\mathbf{u}) d\mathbf{u}$, based on which the information gain is defined as

$$IG(\mathbf{X}_*; \mathbb{P}(\mathbf{u}|\mathbf{y})) := \mathcal{H}[\mathbb{P}(\mathbf{u}|\mathbf{y})] - \int \mathbb{P}(\mathbf{y}_*|\mathbf{y}) \mathcal{H}[\mathbb{P}(\mathbf{u}|\mathbf{y}, \mathbf{y}_*)] d\mathbf{y}_*, \tag{6.1}$$

where the expectation w.r.t. $\mathbb{P}(\mathbf{y}_*|\mathbf{y}) = \int \mathbb{P}(\mathbf{u}|\mathbf{y}) \mathbb{P}(\mathbf{y}_*|\mathbf{u}, \mathbf{y}) d\mathbf{u}$ is required, since the particular outcome $\mathbf{y}_*$ for a candidate $\mathbf{X}_*$ is unknown at scoring time. Neither the posterior $\mathbb{P}(\mathbf{u}|\mathbf{y})$ nor the score values $IG(\mathbf{X}_*; \mathbb{P}(\mathbf{u}|\mathbf{y}))$ can be computed in closed form; they are approximated by a tractable $\mathbb{Q}(\mathbf{u}|\mathbf{y}) = \mathcal{N}(\mathbf{u}|\mathbf{m}, \mathbf{V})$ and $IG(\mathbf{X}_*; \mathbb{Q}(\mathbf{u}|\mathbf{y}))$.

Our sequential algorithm (visualised in figure 6.3) provides a goal-directed way to optimise k-space sampling. In each round, only a single new real measurement is required, while the effective search space, the set of all combinations of candidates, has exponential size in the number of rounds. This characteristic sets it apart from blindly randomised approaches, which explore the search space in stochastic, non-adaptive patterns, and tend to use many more real measurements than rounds. In practise, our algorithmic scheme has to be adjusted to constraints coming from the MR scanner setup.

Up to now, the distributions $\mathbb{P}(\mathbf{u})$ and $\mathbb{P}(\mathbf{y}|\mathbf{u})$ have not been further specified. The next section instantiates our probabilistic model and discusses estimation or reconstruction techniques.

6.2 Probabilistic model

In the following, we introduce the Gaussian likelihood (section 6.2.1) and describe the sparse image prior (section 6.2.2) along with several estimators for reconstruction. Further, we provide background on point spread functions (PSF) for linear and nonlinear reconstructions (section 6.2.3). The probabilistic model is the same as in chapter 5.2, with the exception that the involved variables are defined over the complex numbers rather than the reals. Therefore, at the expense of being slightly redundant, we restate relevant fact to make the section more readable.

6.2.1 Gaussian likelihood and linear reconstruction

Let $\mathbf{u} \in \mathbb{C}^n$ represent the unknown pixelised MR image to be reconstructed, where n is the number of pixels. MR measurements $\mathbf{y}$, linearly depending on the proton density of the object $\mathbf{u}$, (see section 6.1.2) are modelled as

$$\mathbf{y} = \mathbf{X}\mathbf{u} + \boldsymbol{\varepsilon}, \quad \Re(\boldsymbol{\varepsilon}), \Im(\boldsymbol{\varepsilon}) \sim \mathcal{N}(\mathbf{0}, \sigma^2\mathbf{I}), \quad \boldsymbol{\varepsilon} \in \mathbb{C}^m,$$

where $\boldsymbol{\varepsilon}$ accounts for measurement errors, and $z = \Re(z) + \mathrm{i}\Im(z) \in \mathbb{C}$, $\mathrm{i} = \sqrt{-1}$, $[\Re(z), \Im(z)] \in \mathbb{R}^2$. The design or measurement matrix $\mathbf{X} \in \mathbb{C}^{m\times n}$ contains Fourier filters at certain k-space points, and m is the number of k-space measurements taken. Standard linear reconstruction (chapter 2.6), maximises the Gaussian likelihood $\mathbb{P}(\mathbf{y}|\mathbf{u}) = \mathcal{N}(\mathbf{y}|\mathbf{X}\mathbf{u}, \sigma^2\mathbf{I})$ as a function of the bitmap $\mathbf{u}$. The maximum likelihood (ML) or equivalently ordinary least squares (OLS) estimator

$$\hat{\mathbf{u}}_{\mathrm{ML}} = \hat{\mathbf{u}}_{\mathrm{OLS}} = \arg\min_{\mathbf{u}} \|\mathbf{X}\mathbf{u} - \mathbf{y}\|^2 \overset{(i)}{=} \mathbf{X}^+\mathbf{y} \overset{(ii)}{=} \mathbf{X}^{\mathsf{H}}(\mathbf{X}\mathbf{X}^{\mathsf{H}})^{-1}\mathbf{y}$$

is linear in the measurements $\mathbf{y}$ and most appropriate for full-rank measurement designs $\mathbf{X}$. Low-rank designs $\mathbf{X}$ with $m < n$ correspond to undersampling, i.e. reconstruction from incomplete measurements leaving the OLS estimator underdetermined by $m - n$ degrees of freedom. A widely used additional constraint is to select $\mathbf{u}$ with minimal norm as implemented by the pseudo-inverse in equality *(i)*, i.e. $\hat{\mathbf{u}}_{\mathrm{OLS}} = \arg\min\{\|\mathbf{u}\|^2, \mathbf{X}\mathbf{u} = \mathbf{y}\}$. The identity *(ii)* is only sensible for full rank matrices $\mathbf{X}\mathbf{X}^{\mathsf{H}} \in \mathbb{C}^{m\times m}$, $m \leq n$.

In *Cartesian* imaging, k-space is sampled on a rectangular equispaced grid. If all of k-space is acquired, $\mathbf{X}$ equals the orthonormal Fourier matrix $\mathbf{F}$. The estimator $\hat{\mathbf{u}}_{\mathrm{OLS}}$ is obtained by a single inverse FFT since $\mathbf{X}^+ = \mathbf{F}^+ = \mathbf{F}^{-1} = \mathbf{F}^{\mathsf{H}}$. For undersampled Cartesian measurements we have $\mathbf{X} = \mathbf{S}\mathbf{F}$, where $\mathbf{S} \in \{0,1\}^{m\times n}$ is a diagonal selector matrix leading to $\mathbf{X}^+ = \mathbf{F}^{\mathsf{H}}\mathbf{F}^{\mathsf{H}}\mathbf{S}^\top$.

In *spiral* or *radial* imaging, where the measurements are not lying on an equispaced grid, one usually approximates $\mathbf{X} \approx \mathbf{C}\mathbf{F}\mathbf{D}$, where $\mathbf{D}$ is a diagonal weighting matrix and $\mathbf{C}$ is a banded interpolation matrix using Kaiser-Bessel windows [Bernstein et al., 2004, chapter 13.2]. Computation of $\hat{\mathbf{u}}_{\mathrm{OLS}}$ amounts to solving the normal equations $\mathbf{X}^{\mathsf{H}}\mathbf{X}\hat{\mathbf{u}}_{\mathrm{OLS}} = \mathbf{X}^{\mathsf{H}}\mathbf{y}$ by an iterative method like the conjugate gradient based `LSQR` algorithm[2] [Paige and Saunders, 1982]. A simpler linear reconstruction uses the so called zero filling density compensation (ZFDC[3]) [Bernstein et al., 2004, chapter 13.2.4] estimator

$$\hat{\mathbf{u}}_{\mathrm{ZFDC}} = \mathbf{X}^{\mathsf{H}}\mathbf{G}_{\mathbf{k}}\mathbf{y},$$

where $\mathbf{G}$ is a diagonal weighting matrix compensating for sampling density differences in k-space. One commonly uses the area of the tiles of a Voronoi tessellation of k-space, where the centres correspond to the sampling points in k-space to re-weight the measurements $\mathbf{y}$. The

[2] Available from `http://stanford.edu/group/SOL/software/lsqr.html`.

[3] Code at `www.stanford.edu/~mlustig/SparseMRI.html`.

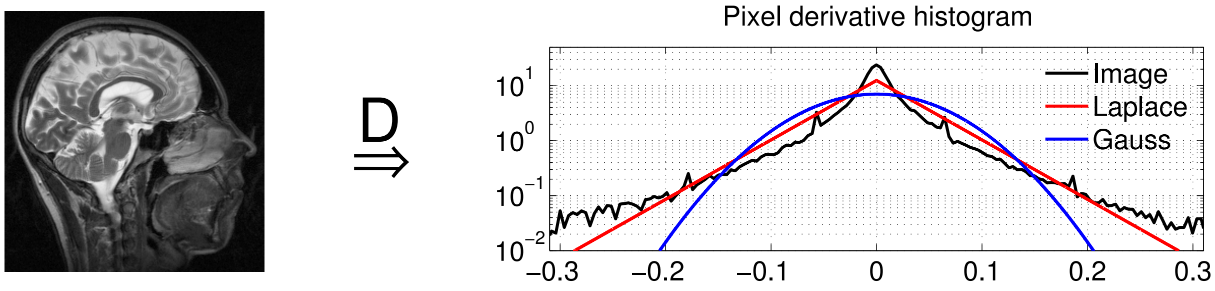


Figure 6.4: Transform sparsity in images
Multiscale gradients of natural and medical images show a particular structure: their histogram has heavy tails and a sharp peak at zero allowing for sharp edges and smooth surfaces simultaneously. For comparison, we show a Gaussian distribution; in our experiments we use the Laplace potential – a tractable compromise.

estimator $\hat{\mathbf{u}}_{\text{ZFDC}}$ can be understood as an approximation to $\hat{\mathbf{u}}_{\text{OLS}}$, where the diagonal matrix $\mathbf{G_k}$ replaces $(\mathbf{X}\mathbf{X}^{\mathsf{H}})^{-1}$.

Neither of the described techniques can deal with undersampled data but all of them are linear in the measurements. Furthermore, $\hat{\mathbf{u}}_{\text{ZFDC}}$ and Cartesian $\hat{\mathbf{u}}_{\text{OLS}}$ are computationally extremely attractive because they require only a single MVM with $\mathbf{X}^{\mathsf{H}}$, which is the main reason, why these two linear reconstruction methods are predominantly used in practise. In order to improve upon the discussed estimation schemes one can take signal class knowledge in form of a *prior* probability distribution $\mathbb{P}(\mathbf{u})$ over bitmaps into account.

6.2.2 Sparsity of MR images and nonlinear reconstruction

A prior is a preference weighting factor, unrelated to the measured data, assigns low density to noise bitmaps and high density to bitmaps in agreement with knowledge about MR images. The vast majority of possible bitmaps do not constitute valid MR images, which are statistically tightly constrained. On a low level, images exhibit sparsity: coefficients $\mathbf{s} = \mathbf{Bu}$ in linear transform spaces have super-Gaussian distributions (see Simoncelli [1999] and figure 6.4). Besides strong pixel correlations, the low entropy of the super-Gaussian distributions are responsible for the high compression rates achieved by modern schemes such as JPEG. Sparsity is a robust property of non-synthetic images, coming from structure (edges, smooth areas, textures) not present in noise. Among many sparsity-enforcing potentials, Laplace potentials

$$\mathcal{T}_j(s_j) \propto e^{-(\tau_j/\sigma)|s_j|}, \quad \tau_j, \sigma > 0$$

with scaling parameters τ_j/σ stand out: they are the best compromise between a close match to natural images statistics (as in figure 6.4) and analytic and algorithmic tractability in inference and estimation. Most prominently, $-\ln \mathcal{T}_j(s_j) \overset{c}{=} (\tau_j/\sigma)|s_j|$ is convex, so that the MAP estimator $\hat{\mathbf{u}}_{\text{MAP}}$ can be computed as a convex program [Tibshirani, 1996].

Our sparse image prior $\mathbb{P}(\mathbf{u})$ collects all super-Gaussian potentials $\mathbb{P}(\mathbf{u}) \propto \prod_{j=1}^{q} \mathcal{T}_j(s_j) = \exp(-\left\|\boldsymbol{\tau} \odot (\mathbf{Bu})\right\|_1 /\sigma)$, where $\mathbf{s} = \mathbf{Bu} \in \mathbb{C}^q$ consists of linear filter responses, with $\mathbf{B} \in \mathbb{R}^{q\times n}$: the image gradient (horizontal and vertical discrete first derivatives; also called total variation coefficients), and coefficients for an orthonormal multi-scale wavelet transform (Daubechies 4, recursion depth 6), a total of $q \approx 3 \cdot n$ Laplace potentials as illustrated in figure 6.5.

The combination of a sparsity prior $\mathbb{P}(\mathbf{u})$ and a Gaussian likelihood $\mathbb{P}(\mathbf{y}|\mathbf{u}) = \mathcal{N}(\mathbf{y}|\mathbf{Xu}, \sigma^2\mathbf{I})$ form a sparse linear model (SLM), due to the linear measurements and the sparsity enforcing prior. Combining these two terms by Bayes' rule, we have

$$\mathbb{P}(\mathbf{u}|\mathbf{y}) \propto \mathbb{P}(\mathbf{y}|\mathbf{u})\mathbb{P}(\mathbf{u}),$$

where $\mathbb{P}(\mathbf{u}|\mathbf{y})$ is the Bayesian posterior distribution, the canonical combination of measurement data and prior knowledge by rules of probability. Both prior $\mathbb{P}(\mathbf{u})$ and posterior $\mathbb{P}(\mathbf{u}|\mathbf{y})$

$$\mathbb{P}(\underline{\mathbf{u}}) \propto \prod_{j=1}^{q} \mathcal{T}_j(s_j) = \exp\left(-\tfrac{\tau_D}{\sigma}\,\|\underline{\mathbf{Du}}\|_1\right) \times\ \exp\left(-\tfrac{\tau_W}{\sigma}\,\|\underline{\mathbf{Wu}}\|_1\right)$$

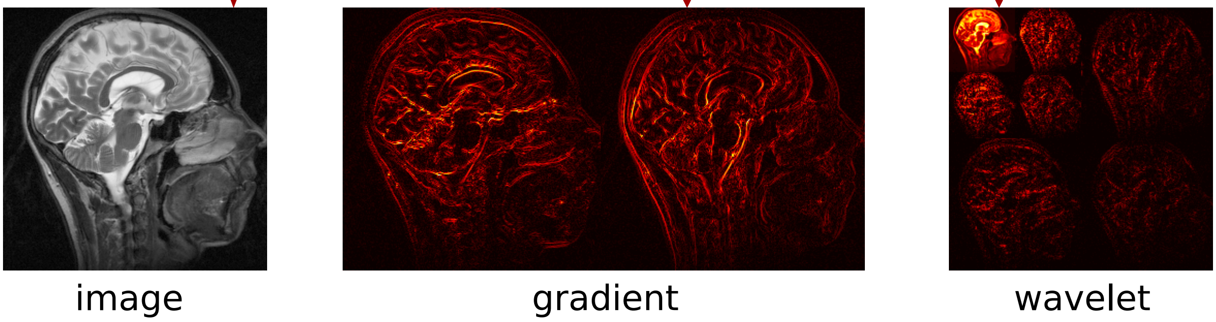


Figure 6.5: Sparsity prior on MR image
Both finite differences and wavelet coefficients of natural and also medical images are sparse. Therefore, our image prior encodes precisely that low-level information. The wavelet transform $\mathbf{W}$ *can be understood as gradient on larger scales. Therefore,* $\mathbf{B} = [\mathbf{D}^\top, \mathbf{W}^\top]^\top$ *computes multiscale derivatives of the image and our prior can be seen as favouring mainly smooth images containing occasionally some edges.*

are distributions over bitmaps, representing our knowledge about the image before and after measurements have been obtained. In sparse reconstruction techniques, the posterior is optimised, instead of the likelihood alone. The prominent MAP (maximum a posteriori) estimation algorithm, seeks the mode of the posterior

$$\hat{\mathbf{u}}_{\mathrm{MAP}} = \arg\max_{\mathbf{u}} \mathbb{P}(\mathbf{u}|\mathbf{y}) = \arg\min_{\mathbf{u}} \{-\ln \mathbb{P}(\mathbf{y}|\mathbf{u}) - \ln \mathbb{P}(\mathbf{u})\}, \tag{6.2}$$

which exactly corresponds to the sparse reconstruction method of Lustig et al. [2007]. In order to favour a $\mathbf{u}$ that is close to real-valued, we make use of n additional Laplace potentials on $\Im(s_j)$, as in Block et al. [2007], but not in Lustig et al. [2007]. Since $s_j \in \mathbb{C}$ is represented by $[\Re(s_j), \Im(s_j)] \in \mathbb{R}^2$ internally, this amounts to a simple extension of $\mathbf{B}$. The MAP reconstruction process, which is nonlinear due to the non-Gaussian prior $\mathbb{P}(\mathbf{u})$, is illustrated in Lustig et al. [2007, figure 2]. As opposed to the maximum likelihood estimator, $\hat{\mathbf{u}}_{\mathrm{MAP}}$ cannot be found by a single linear system, but requires iterative computation. In our SLM setting, it is the unique minimiser of a convex criterion, and efficient MAP algorithms are available [Chen et al., 1999, Tibshirani, 1996].

In the context of reconstruction of undersampled signals $m < n$, the prior imposes a structure on the space of possible signals $\mathbf{u}$ that could have generated the measurements $\mathbf{y}$. Consequently, prior knowledge allows to reconstruct MR images from measurements far below the Nyquist limit.

In our experiments, scale parameters τ_j are shared among all potentials of the same kind, but we allow for different values in wavelet coefficient, total variation, and imaginary part potentials. While Bayesian inference is approximated over primary parameters $\mathbf{u}$, hyperparameters τ_j, σ^2 are estimated in general. In our experiments, we optimised them on data not used for comparisons, then fixed these values for all subsequent sampling optimisation and MAP reconstruction runs. We selected the τ_W/τ_D scale parameters optimally for the Nyquist spiral $\mathbf{X}_{\mathrm{nyq}}$, and set σ^2 to the variance of $\mathbf{X}_{\mathrm{nyq}}(\mathbf{u}_{true} - |\mathbf{u}_{true}|)$.

6.2.3 Point spread functions and experimental design

The concept of a point spread function (PSF) or impulse response function $\hat{\mathbf{e}}_i$ is a very helpful tool to describe and analyse *linear* (imaging) systems. Denote the imaging system by ς, the object under investigation by $\mathbf{u} \in \mathbb{C}^n$ and the estimated image by $\hat{\mathbf{u}} \in \mathbb{C}^n$. By virtue of linearity, the outcome of a linear superposition of objects $\mathbf{u}_1$ and $\mathbf{u}_2$ equals the superposition of the individual estimates $\hat{\mathbf{u}}_1$ and $\hat{\mathbf{u}}_2$

$$\varsigma(\lambda_1\mathbf{u}_1 + \lambda_2\mathbf{u}_2) \quad = \quad \lambda_1\varsigma(\mathbf{u}_1) + \lambda_2\varsigma(\mathbf{u}_2) = \lambda_1\hat{\mathbf{u}}_1 + \lambda_2\hat{\mathbf{u}}_2.$$

Estimator	Symbol $\hat{\mathbf{u}} = \mathbf{Ry}$	Reconstruction operator $\mathbf{R}$	
		Cartesian	Non-Cartesian
Maximum likelihood, ordinary least squares	$\hat{\mathbf{u}}_{\text{ML}} = \hat{\mathbf{u}}_{\text{OLS}}$	$\mathbf{F}^{\mathsf{H}}\mathbf{S}^{\top}$	$\mathbf{X}^{\mathsf{H}}(\mathbf{X}\mathbf{X}^{\mathsf{H}})^{-1}$
Zero filling density compensation	$\hat{\mathbf{u}}_{\text{ZFDC}}$	$\mathbf{F}^{\mathsf{H}}\mathbf{S}^{\top}\mathbf{G}_{\mathbf{k}}$	$\mathbf{X}^{\mathsf{H}}\mathbf{G}_{\mathbf{k}}$
Variational Bayesian mean	$\hat{\mathbf{u}}_{\text{VB}}$	$\mathbf{R}^{\text{VB}}_{\mathbf{y}} = (\mathbf{X}^{\mathsf{H}}\mathbf{X} + \mathbf{B}^{\mathsf{H}}\mathbf{\Gamma}_{\mathbf{y}}^{-1}\mathbf{B})^{-1}\mathbf{X}^{\mathsf{H}}$	
Maximum a posteriori, penalised least squares, $p = 2$	$\hat{\mathbf{u}}_{\text{MAP}} = \hat{\mathbf{u}}_{\text{PLS}}$	$\left(\mathbf{X}^{\mathsf{H}}\mathbf{X} + \gamma^{-1}\mathbf{B}^{\mathsf{H}}\mathbf{B}\right)^{-1}\mathbf{X}^{\mathsf{H}}$	
Maximum a posteriori, penalised least squares, $p \neq 2$	$\hat{\mathbf{u}}_{\text{MAP}} = \hat{\mathbf{u}}_{\text{PLS}} = \arg\min_{\mathbf{u}} \lVert\mathbf{Xu} - \mathbf{y}\rVert_2^2 + \gamma^{-1}\lVert\mathbf{Bu}\rVert_p^p$		

Table 6.1: Reconstruction operators for different estimators

Furthermore, linear systems ς can be represented by a matrix $\mathbf{S} \in \mathbb{C}^{n \times n}$: $\hat{\mathbf{u}} = \varsigma(\mathbf{u}) = \mathbf{Su}$. In our image measurement application, $\mathbf{S}$ naturally decomposes into a measurement $\mathbf{X}$ and reconstruction operator $\mathbf{R}$, $\mathbf{S} = \mathbf{RX}$. A system $\mathbf{S}$ is linear if and only if $\mathbf{R}$ does not depend on $\mathbf{y}$. Table 6.1 summarises several linear and nonlinear reconstruction operators frequently used in MRI reconstruction as introduced in sections 6.2.1 and 6.2.2. We normalise the rows $\mathbf{x}_i$ of $\mathbf{X}$ to unit length $\lVert\mathbf{x}_i\rVert = 1$ to remove the scaling ambiguity between $\mathbf{R}$ and $\mathbf{X}$ if $\mathbf{S}$ is kept constant. Now, we can represent the underlying object $\mathbf{u} = \sum_{i=1}^{n} u_i \mathbf{e}_i$ in the standard basis with unit vectors $\mathbf{e}_i$, apply the system function ς and use linearity to see that the estimated image

$$\hat{\mathbf{u}} = \varsigma(\mathbf{u}) = \varsigma\left(\sum_{i=1}^{n} u_i \mathbf{e}_i\right) = \sum_{i=1}^{n} u_i \varsigma(\mathbf{e}_i) = \sum_{i=1}^{n} u_i \hat{\mathbf{e}}_i$$

is a weighted sum of impulse responses or point spread functions $\hat{\mathbf{e}}_i$ independent of the measurements $\mathbf{y}$. To understand what the imaging system ς is doing, one simply needs to know all PSFs. Furthermore, the quality or resolution of an imaging system can be quantified by the deviation $\Delta(\mathbf{X}) = \sum_i \lVert\hat{\mathbf{e}}_i(\mathbf{X}) - \mathbf{e}_i\rVert$ – a perfect imaging system has $\mathbf{R} = \mathbf{X}^{-1} \Leftrightarrow \mathbf{S} = \mathbf{I}$ and therefore no resolution is lost since $\mathbf{u} = \hat{\mathbf{u}}$. More precisely, the off-diagonal elements $\mathbf{S}$ can be used to quantify, how much resolution is lost. As a result of the linearity, the measurement process $\mathbf{X}$ and reconstruction process $\mathbf{R}$ do not depend on the signal $\mathbf{u}$; the system does not distinguish between random noise inputs and proper MR images. Furthermore, undersampling $\mathbf{X} \in \mathbb{C}^{m \times n}$ with $m < n$ necessarily leads to a loss in resolution because $\mathbf{X}$ cannot be inverted.

The deviation $\Delta(\mathbf{X}, \mathbf{X}_*)$ is a measure for the resolution of the imaging system. Consequently selecting $\mathbf{X}_*$ according to $\Delta(\mathbf{X}, \mathbf{X}_*)$ is a very promising criterion for experimental design.

Nonlinear systems, however, are much more difficult to characterise because their behaviour can be qualitatively different depending on the input. In MRI imaging systems ς, the noisy measurement $\mathbf{y}$ are linearly related to $\mathbf{u}$ by the measurement design $\mathbf{X}$, however the reconstruction $\hat{\mathbf{u}} = \boldsymbol{\rho}(\mathbf{y})$ can be nonlinear. Both the MAP estimator

$$\hat{\mathbf{u}}_{\text{MAP}} = \boldsymbol{\rho}_{\text{MAP}}(\mathbf{y}) = \arg\min_{\mathbf{u}} \lVert\mathbf{Xu} - \mathbf{y}\rVert_2^2 + \gamma^{-1}\lVert\mathbf{Bu}\rVert_p^p, \; p \neq 2$$

(section 6.2.2 and table 6.1) and the VB (variational Bayesian) mean estimator

$$\hat{\mathbf{u}}_{\text{VB}} = \boldsymbol{\rho}_{\text{VB}}(\mathbf{y}) = \mathbf{R}^{\text{VB}}_{\mathbf{y}}\mathbf{y} = \left(\mathbf{X}^{\mathsf{H}}\mathbf{X} + \mathbf{B}^{\mathsf{H}}\mathbf{\Gamma}_{\mathbf{y}}^{-1}\mathbf{B}\right)^{-1}\mathbf{X}^{\mathsf{H}}\mathbf{y}, \; \gamma_{\mathbf{y}} = \arg\min_{\gamma} \phi(\gamma, \mathbf{y})$$

(section 6.3 and table 6.1) are nonlinear reconstructions $\boldsymbol{\rho}(\mathbf{y})$ rendering the entire imaging system nonlinear. The PSFs $\hat{\mathbf{e}}_i = \boldsymbol{\rho}(\mathbf{X}\mathbf{e}_i)$ (they are called transform point spread functions in Lustig et al. [2007]) do not satisfactorily characterise the system since they depend nonlinearly on the measurements. For example, in the SLM, we have $\hat{\mathbf{u}}_{\text{MAP}} = \mathbf{u}$ for many piecewise constant images [Lustig et al., 2007], whereas random noise bitmaps $\mathbf{u}$ are very unfaithfully reconstructed. In contrast to the linear case, this renders the deviation $\Delta(\mathbf{X}, \mathbf{x}_*)$ useless. Therefore, we will use the information gain criterion $IG(\mathbf{X}_*; \mathbb{P}(\mathbf{u}|\mathbf{y}))$ of section 6.1.3 for experimental design.

6.3 Variational inference

In order to compute design score values $IG(\mathbf{X}_*; \mathbb{P}(\mathbf{u}|\mathbf{y}))$, we have to integrate over the posterior $\mathbb{P}(\mathbf{u}|\mathbf{y})$. These computations, referred to as Bayesian *inference*, cannot be done exactly in the case of sparse linear models. We use the algorithm of chapter 3.5 for SLM approximate inference, which scales up to high-resolution MR images, while being accurate enough to successfully drive nonlinear design optimisation. The intractable posterior $\mathbb{P}(\mathbf{u}|\mathbf{y})$ is fitted by a Gaussian distribution $\mathbb{Q}(\mathbf{u}|\mathbf{y})$, with the aim of closely approximating the posterior mean and covariance matrix. Fitting amounts to a convex optimisation problem with a unique solution.

In the following, we discuss the general idea of the inference procedure, then we look into details of the optimisation, especially the Lanczos marginal variance estimation. Finally, we discuss sparse and least squares estimation as special cases of variational Bayesian mean estimation and reveal insightful links between the estimation techniques.

6.3.1 Highlevel overview

We employ the *variational* relaxation introduced in chapter 3 because the associated algorithm is scalable. The posterior $\mathbb{P}(\mathbf{u}|\mathbf{y})$ is approximated by the closest Gaussian distribution $\mathbb{Q}(\mathbf{u}|\mathbf{y})$ from a large approximation family. Since integrations against Gaussian densities are tractable even in high dimensions, the replacement $\mathbb{P}(\mathbf{u}|\mathbf{y}) \to \mathbb{Q}(\mathbf{u}|\mathbf{y})$ allows for design score computations on a large scale.

Our prior $\mathbb{P}(\mathbf{u})$ as discussed in section 6.2.2 is a product of super-Gaussian Laplace potentials, each of which can be tightly lower bounded by Gaussian functions of any variance (see figure 3.2). We use this property to choose the approximation family, and to formulate the variational problem. For the former, we start with $\mathbb{P}(\mathbf{u}|\mathbf{y})$, but replace each prior potential by a Gaussian lower bound centred at zero. The variances $\gamma = [\gamma_j]_j \in \mathbb{R}^q_+$ of these replacements parametrise the Gaussian family members $\mathbb{Q}(\mathbf{u}|\mathbf{y};\gamma)$. For the variational criterion $\phi(\gamma)$, we apply the same replacement to the log partition function

$$\ln \mathbb{P}(\mathbf{y}) = \ln \int \mathbb{P}(\mathbf{y}|\mathbf{u})\mathbb{P}(\mathbf{u})d\mathbf{u}, \tag{6.3}$$

the approximation target in most variational inference methods (posterior moments, such as mean and covariance, are obtained as derivatives of $\ln \mathbb{P}(\mathbf{y})$) [Jordan, 1997], leaving us with a lower bound $-\phi(\gamma)/2 \leq \ln \mathbb{P}(\mathbf{y})$, which can be evaluated as a Gaussian integral. The larger the lower bound, the tighter is the fit of $\mathbb{Q}(\mathbf{u}|\mathbf{y})$ to $\mathbb{P}(\mathbf{u}|\mathbf{y})$ since $2\phi(\gamma) + \ln \mathbb{P}(\mathbf{y})$ is a convex upper bound to the Kullback-Leibler divergence $\mathrm{KL}[\mathbb{Q}(\mathbf{u}|\mathbf{y}) \,\|\, \mathbb{P}(\mathbf{u}|\mathbf{y})]$, a standard measure for the difference between two distributions [Cover and Thomas, 2006].

We established in chapter 3.4 that the variational inference problem $\min_\gamma \phi(\gamma)$ is convex: there is a single best Gaussian fit $\mathbb{Q}(\mathbf{u}|\mathbf{y})$ to $\mathbb{P}(\mathbf{u}|\mathbf{y})$. Moreover, we proposed a double loop algorithm to find the minimum point of ϕ, rapid enough to address the k-space optimisation problem. Revisiting algorithm 6.1, we obtain our method in practise by replacing $\mathbb{P}(\mathbf{u}|\mathbf{y}) \to \mathbb{Q}(\mathbf{u}|\mathbf{y})$, which is fitted before starting the design loop, and refitted to the extended posterior at the end of each round, in step (4). The optimisation is reduced to calling primitives of numerical computing a moderate number of times: reweighted least squares estimation, and approximate eigendecomposition. While the former is routinely used for linear and nonlinear MRI reconstruction, the latter seems specific to the inference problem and is required in order to approximate posterior covariances. These are further reduced, by standard algorithms of numerical mathematics, to signal processing primitives such as fast Fourier transform (FFT) or non-equispaced fast Fourier transform (NFFT).

6.3.2 Experimental design details

Once $\mathbb{P}(\mathbf{u}|\mathbf{y})$ is replaced by its closest Gaussian fit $\mathbb{Q}(\mathbf{u}|\mathbf{y};\gamma)$, the design score (6.1) can be computed (step (1) in algorithm 6.1). However, k-space optimisation comes with large candidate

$$\phi(\gamma) = \overbrace{\begin{array}{c} \boxed{\text{Outer loop:}} \\ \mathbf{z}\leftarrow \mathrm{dg}\left(\mathbf{B}\mathbf{A}_\gamma^{-1}\mathbf{B}^\top\right),\quad a) \\ \overbrace{\min_{\mathbf{z}\succeq\mathbf{0}} \mathbf{z}^\top\gamma^{-1} - \phi_\cap^*(\mathbf{z})} \\ \shortparallel \\ \underbrace{\ln|\mathbf{A}_\gamma|}_{\substack{\phi_\cap(\gamma^{-1}) \text{ concave in } \gamma^{-1}\\ \text{also convex in } \gamma}}\end{array} + \begin{array}{c} \mathbf{A}_\gamma = \mathbf{X}^{\mathsf{H}}\mathbf{X}+\mathbf{B}^\top\boldsymbol{\Gamma}^{-1}\mathbf{B} \\ \overbrace{\gamma^\top\boldsymbol{\tau}^2 - \sigma^{-2}\mathbf{y}^{\mathsf{H}}\mathbf{X}\mathbf{A}_\gamma^{-1}\mathbf{X}^{\mathsf{H}}\mathbf{y}+\sigma^{-2}\mathbf{y}^{\mathsf{H}}\mathbf{y}} \\ \shortparallel \\ \underbrace{\gamma^\top\boldsymbol{\tau}^2+\sigma^{-2}\min_{\mathbf{u}}\|\mathbf{y}-\mathbf{X}\mathbf{u}\|^2+\mathbf{s}^{\mathsf{H}}\boldsymbol{\Gamma}^{-1}\mathbf{s}}_{\phi_\cup(\gamma)\text{ convex in }\gamma}\end{array}}^{\begin{array}{l}\boxed{\text{Inner loop:}}\quad \gamma \leftarrow (\sigma\boldsymbol{\tau})^{-1}\odot\sqrt{\sigma^2\mathbf{z}+\mathbf{s}^2},\ \mathbf{s}=\mathbf{B}\mathbf{u}_* \quad b) \\ \qquad\qquad\qquad \mathbf{u}_* \leftarrow \arg\min_{\mathbf{u}} \frac{1}{2\sigma}\|\mathbf{y}-\mathbf{X}\mathbf{u}\|^2+\boldsymbol{\tau}^\top\sqrt{\sigma^2\mathbf{z}+\mathbf{s}^2} \quad c)\end{array}}$$

Figure 6.6: Double loop variational inference algorithm for MRI
In approximate inference, the (convex) variational criterion $\phi(\gamma)$ is minimised by decomposing it into a coupled $\phi_\cap(\gamma^{-1})$ and a decoupled part $\phi_\cup(\gamma)$. The coupled part is concave and can therefore be upper bounded by a (decoupled) linear function a) leading to the outer loop step of the algorithm. The (decoupled) surrogate objective $\phi_{\mathbf{z}}(\gamma, \mathbf{u}) = \mathbf{z}^\top\gamma^{-1} + \phi_\cup(\gamma)$ is minimised in the inner loop b) + c). First, the minimisation in γ can be done analytically b) leaving us with a penalised least squares problem c). We iterate between inner and outer loop updates until convergence.

elements $\mathbf{X}_* \in \mathbb{C}^{d\times n}$ (the spiral interleaves used in our study consist of $d = 3216$ k-space points, hence $\mathbf{X}_* \in \mathbb{C}^{3216\times n}$, $n = 256^2$), and if many of these candidates are to be scored in each round, a naïve computation is too slow. For our score computation, we make use of the approximate eigendecomposition once more.

From the fitted distribution $\mathbb{Q}(\mathbf{u}|\mathbf{y};\gamma)$, we compute design scores $IG(\mathbf{X}_*;\mathbb{Q}(\mathbf{u}|\mathbf{y};\gamma))$ by noting that $\mathcal{H}[\mathbb{Q}(\mathbf{u}|\mathbf{y};\gamma)] = \frac{1}{2}\log|2\pi e\sigma^2\mathbf{A}^{-1}|$, so that $IG(\mathbf{X}_*;\mathbb{Q}(\mathbf{u}|\mathbf{y})) = \log|\mathbf{I}+\mathbf{X}_*\mathbf{A}^{-1}\mathbf{X}_*^{\mathsf{H}}|$ (see chapter 2.6.2). Here, we approximate $\mathbb{P}(\mathbf{u}|\mathbf{y},\tilde{\mathbf{y}}_*)$ by $\propto \mathbb{Q}(\mathbf{u}|\mathbf{y};\gamma)\mathbb{P}(\tilde{\mathbf{y}}_*|\mathbf{u})$ *without* refitting the variational parameters γ. If $\mathbf{X}_* \in \mathbb{C}^{d\times n}$, $IG(\mathbf{X}_*)$ can be computed by solving d linear systems, but this is too slow to be useful. Instead, we use the Lanczos approximate eigendecomposition once more: $\ln|\mathbf{I}+\mathbf{X}_*\mathbf{A}^{-1}\mathbf{X}_*^{\mathsf{H}}| \approx \ln|\mathbf{I}+\mathbf{V}_*^{\mathsf{H}}\mathbf{V}_*|$, $\mathbf{V}_* := \mathbf{\Lambda}^{-1/2}\mathbf{Q}^{\mathsf{H}}\mathbf{X}_*^{\mathsf{H}} \in \mathbb{C}^{k\times d}$. If $k < d$, we compute $\ln|\mathbf{I}+\mathbf{V}_*\mathbf{V}_*^{\mathsf{H}}|$ instead. This approximation allows us to score many large candidates in each design round. Moreover, the score computation can readily be parallelised across different processors or machines. We compared approximate score values to true ones, on 64 × 64 images where the latter can be computed. While the true values were strongly underestimated in general (even the largest ones), the peaks of the score curves were traced correctly by the approximations, and the maximisers of the approximate curves fell within dominating peaks of the exact score.

6.3.3 Inference algorithm details

Our double loop algorithm to minimise the variational criterion $\phi(\gamma)$ is a special case of algorithm 3.1, where all potentials are Laplace and the $\phi_\cup^{(2)}$ bound is used (see chapter 3.5.3 for details). Figure 6.6 summarises how we iterate between inner and outer loops in order to solve the variational problem. An equivalent but more detailed picture is provided in algorithm 6.2.

Approximate inference is used at different points in algorithm 6.1: in the initial phase before the design loop, and at the end of each round. In our experiments, we used 5 outer loop steps in the initial phase, and a single outer loop step between design extensions. We ran up to 30 inner loop IRLS steps, with up to 750 LCG iterations for each linear system (they often converged much faster). To save time, we partitioned the IRLS steps in categories "sloppy" and "convergence". Sloppy steps use 150 LCG iterations only, preceding convergence steps.

***Algorithm 6.2** Double loop variational inference algorithm for MRI*

```
Require: Data $\mathbf{X}, \mathbf{y}$
  $\boxed{\textbf{Outer loop:}}$ marginal variances $\boldsymbol{\nu} = \mathrm{dg}\left(\mathbf{B}\mathbb{V}_{\mathrm{Q}(\mathbf{u}|\mathcal{D})}[\mathbf{u}]\mathbf{B}^\top\right) = \sigma^2\mathbf{z}$ by Lanczos (chapter 3.5.4)
  Approximate eigendecomposition using $k$-step Lanczos: $\mathbf{A}_{\boldsymbol{\gamma}} = \mathbf{X}^{\mathsf{H}}\mathbf{X} + \mathbf{B}^\top\boldsymbol{\Gamma}^{-1}\mathbf{B} \approx \mathbf{Q}\mathbf{T}\mathbf{Q}^{\mathsf{H}}$
  Refit upper bound $\phi_{\mathbf{z}}(\boldsymbol{\gamma}, \mathbf{u}) = \mathbf{z}^\top\boldsymbol{\gamma}^{-1} + \boldsymbol{\gamma}^\top\boldsymbol{\tau}^2 + \sigma^{-2}\left(\|\mathbf{y}-\mathbf{X}\mathbf{u}\|^2 + \mathbf{s}^\top\boldsymbol{\Gamma}^{-1}\mathbf{s}\right)$ of equation 3.10
  repeat
    $\mathbf{w}_j \leftarrow \frac{1}{\sqrt{\lambda_j}}\mathbf{B}\mathbf{Q}\mathbf{v}_j$, where $\mathbf{V}\boldsymbol{\Lambda}\mathbf{V}^\top = \mathbf{T}$, $\mathbf{z} \leftarrow \sum_{j=1}^{k}\mathbf{w}_j \odot \mathbf{w}_j$
    $\boxed{\textbf{Inner loop:}}$ marginal means $\mathbf{u}_* = \mathbb{E}_{\mathrm{Q}(\mathbf{u}|\mathcal{D})}[\mathbf{u}]$ by IRLS (chapter 3.5.5)
    Find $\mathbf{u}_* \leftarrow \arg\min_{\mathbf{u}} \frac{1}{2\sigma}\|\mathbf{y}-\mathbf{X}\mathbf{u}\|^2 + \boldsymbol{\tau}^\top\sqrt{\sigma^2\mathbf{z}+\mathbf{s}^2}$ of equation 3.12
    repeat
      $\boldsymbol{\varsigma} \leftarrow \sqrt{\sigma^2\mathbf{z}+\mathbf{s}^2}$, $\mathbf{g} \leftarrow \mathbf{X}^{\mathsf{H}}(\mathbf{X}\mathbf{u}-\mathbf{y}) + \sigma\mathbf{B}^\top\boldsymbol{\tau}\odot\mathbf{s}\odot\boldsymbol{\varsigma}^{-1}$, $\mathbf{H} \leftarrow \mathbf{X}^{\mathsf{H}}\mathbf{X} + \sigma^3\mathbf{B}^\top\mathrm{dg}(\boldsymbol{\tau}\odot\mathbf{z}\odot\boldsymbol{\varsigma}^{-3})\mathbf{B}$
      Solve linear system $-\mathbf{H}\mathbf{d} \leftarrow \mathbf{g}$ by CG to obtain Newton direction $\mathbf{d}$
      Find step size $\lambda$ by line search along $\phi_{\mathbf{z}}(\mathbf{u}+\lambda\mathbf{d})$, update $\mathbf{u} \leftarrow \mathbf{u}+\lambda\mathbf{d}$
    until Inner loop converged
    Update $\mathbf{s} = \mathbf{B}\mathbf{u}_*$, $\boldsymbol{\gamma} \leftarrow (\sigma\boldsymbol{\tau})^{-1} \odot \sqrt{\sigma^2\mathbf{z}+\mathbf{s}^2}$
  until Outer loop converged
```

Double loop variational inference algorithm for the special case of the sparse linear model with Laplace potentials. The objective $\phi(\boldsymbol{\gamma}, \mathbf{u})$ of equation 3.6 is jointly minimised w.r.t. $\boldsymbol{\gamma}$ and $\mathbf{u}$ by refitting an auxiliary upper bound $\phi_{\mathbf{z}}(\boldsymbol{\gamma}, \mathbf{u})$ in every outer loop iteration, which is then minimised in the inner loop by a Newton algorithm.

The Lanczos algorithm was run for $k = 750$ iterations in general.

The approximate computation of the marginal variances $\boldsymbol{\nu}$ in the outer loop using the Lanczos algorithm is a crucial step. As mentioned in chapter 3.5.6 and detailed in the next section, underestimated marginal variances bias the model towards MAP estimation. Following the analysis in chapter 3.2 and the related figure 3.5, we analyse Lanczos vector convergence and variance estimation errors in figure 6.7 using a realistic toy model with 32×32 pixels. Importantly, we see that after $k = 200$ Lanczos steps, eigenvalues converged both on the lower and the upper half of the spectrum. The current 200-dimensional approximate eigensystem, however contains also much overlap with exact eigenvectors in the middle of the spectrum as shown in figure 6.7 middle. Furthermore, the marginal variances $\boldsymbol{\nu} = \sigma^2\mathbf{z}$ are heavily underestimated by $\hat{\boldsymbol{\nu}} = \sigma^2\hat{\mathbf{z}}_k$ but interestingly, the relative accuracy for the largest and the smallest variances is higher than for the intermediate ones.

6.3.4 Insights and special cases

Further analytically instructive insights (similar to chapter 3.5.6) can be obtained by looking at some limiting cases of the surrogate (upper bound on the variational) objective function used in the inner loop $\phi_{\mathbf{z}}(\mathbf{u}) = \min_{\boldsymbol{\gamma}} \phi_{\mathbf{z}}(\boldsymbol{\gamma}, \mathbf{u}) =$

$$\min_{\boldsymbol{\gamma}}\left(\mathbf{z}^\top\boldsymbol{\gamma}^{-1} + \boldsymbol{\gamma}^\top\boldsymbol{\tau}^2 + \sigma^{-2}\|\mathbf{y}-\mathbf{X}\mathbf{u}\|^2 + \sigma^{-2}\mathbf{s}^\top\boldsymbol{\Gamma}^{-1}\mathbf{s}\right),\ \boldsymbol{\gamma}_* = (\sigma\boldsymbol{\tau})^{-1} \odot \sqrt{\sigma^2\mathbf{z}+\mathbf{s}^2}.$$

Recall that our variational approximate inference algorithms fits a sequence of Gaussians to a non-Gaussian model, where $\phi_{\mathbf{z}}(\boldsymbol{\gamma}, \mathbf{u})$ serves as goodness-of-fit criterion. In the outer loop, $\mathbf{z}$ is chosen to equal the slope of concave part in the objective in order to optimally upper bound it. Interestingly, for differently chosen $\mathbf{z}$, we still obtain an upper bound on the objective but also converge to a different stationary point.

For particular choices of $\mathbf{z}$, we find that three different estimators $\hat{\mathbf{u}}$: PLS $p = 1$ or MAP, OLS and PLS $p = 2$ emerge as special cases of our criterion.

1. Choosing $\mathbf{z} = \mathbf{0}$ leads to $\boldsymbol{\gamma}_* = (\sigma\boldsymbol{\tau})^{-1} \odot |\mathbf{s}|$ and hence MAP estimation when optimising $\phi_{\mathbf{z}}(\mathbf{u})$.

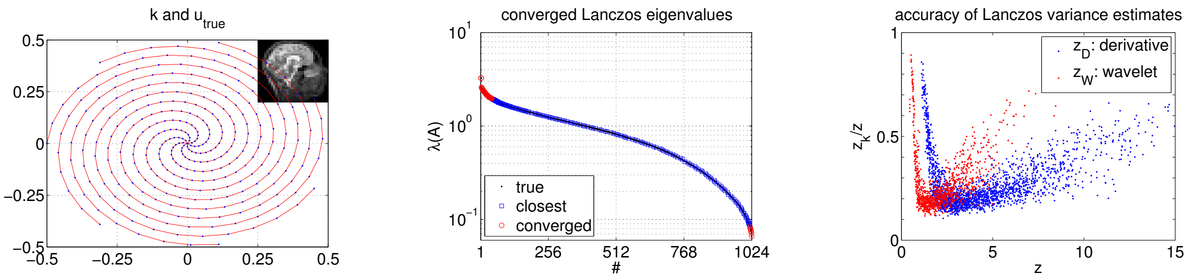


Figure 6.7: Convergence of Lanczos eigenvalues and variance estimation
Left: small scale example with $\mathbf{u} \in \mathbb{C}^{32\times 32}$, $\mathbf{X} \in \mathbb{C}^{672\times 1024}$, $\mathbf{B} \in \mathbb{C}^{3008\times 1024}$, $\sigma^2 = 10^{-5}$, $\tau_W = \tau_D = 0.3$. *We show* $\mathbf{u}_{true}$ *and* $\mathbf{k}$ *the* k*-space coordinates. Middle: convergence of the eigenvalue/eigenvector pairs after* $k = 200$ *Lanczos iterations. The variational parameter* γ *has undergone two loops of the double loop algorithm. Right: relative accuracy of the Lanczos estimate* $\hat{\mathbf{z}}_k$ *compared to* $\mathbf{z}$ *for* $k = 200$.
Middle: eigenvalues converge from top and bottom, the bulk of the vectors in $\mathbf{Q}$ *deals with directions of intermediate eigenvalues. Right: smaller values of* $\mathbf{z}$ *are approximated more accurately; marginal variances of wavelet potentials tend to be smaller.*

2. Set $\mathbf{z} = v^2\mathbf{1}$, then for large v values $\boldsymbol{\gamma}_* \to v\boldsymbol{\tau}^{-1}$. Hence, we have that the terms $\mathbf{z}^\top\boldsymbol{\gamma}^{-1} + \boldsymbol{\gamma}^\top\boldsymbol{\tau}^2 \to v\mathbf{2}^\top\boldsymbol{\tau} = \text{const.}$ and $\mathbf{s}^\top\boldsymbol{\Gamma}^{-1}\mathbf{s} \to v^{-1}\mathbf{s}^\top(\boldsymbol{\tau}\odot\mathbf{s}) \to 0$ are eliminated from $\phi_{\mathbf{z}}(\boldsymbol{\gamma}, \mathbf{u})$ leaving $\sigma^{-2}\left\|\mathbf{y} - \mathbf{X}\mathbf{u}\right\|^2$, which coincides with OLS estimation.

3. Picking $\mathbf{z} = (\rho^2\mathbf{1} - \mathbf{s}^2)/\sigma^2 \succeq \mathbf{0}$ with $\rho > \max_j |s_j|$ and $\boldsymbol{\tau} = \frac{\rho}{\sigma\gamma}\cdot\mathbf{1}$ in a data dependent way, yields $\boldsymbol{\gamma}_* = \gamma\mathbf{1} = \text{const.}$ and hence PLS estimation with $p = 2$.

Besides being formally interesting, these facts show that sparse MAP estimation (1. $\mathbf{z} = \mathbf{0}$) and simple least squares estimation (2. $\mathbf{z} \to \infty$) can be regarded as two ends of the same spectrum with our variational approximation to the posterior mean in between. Sufficient tweaking of the scale parameters $\boldsymbol{\tau}$ (as done in 3.) allows even to obtain the quadratically penalised least squares estimator.

6.4 Experiments

We consider design problems for Cartesian and spiral sequences. In either case, we extract or interpolate measurements corresponding to desired trajectories from scanner data recorded on an equispaced grid (Magneton Trio scanner, Siemens Medical Solutions, Erlangen, Germany; turbo spin echo (TSE) sequence, 23 echos per excitation, train of 120° refocusing pulses, each phase encoded differently, $1 \times 1 \times 4\,\text{mm}^3$; different echo times and orientations, see figure 6.9). Reconstructions $\hat{\mathbf{u}}$ are validated by the L_2 distance $\|\mathbf{u}_{\text{true}} - |\hat{\mathbf{u}}|\|_2$, $\mathbf{u}_{\text{true}}$ being the absolute value of the complete data reconstruction. We use sparse MAP reconstruction in general (equation 6.2), with code as used in Lustig et al. [2007], comparing against linear ZFDC reconstruction (zero filling with density compensation) [Bernstein et al., 2004, chapter 13.2.4] for Cartesian undersampling.

The near-Hermitian structure of measurements is an important instance of prior knowledge, in that samples at $\mathbf{k}$ and $-\mathbf{k}$ are highly redundant. This knowledge is exploited in half-Fourier acquisition [Bernstein et al., 2004, chapter 13.4]. It is built into our model through the real-valuedness of $\mathbf{u}$. In Cartesian sequences, only the upper or lower half of phase encodes is measured, except for a central symmetric slab. For spiral trajectories, we restrict ourselves to offset angles $\theta_0 \in [0, \pi)$. These restrictions do *not* apply to image reconstruction. For MAP, we follow the common practise of reconstructing a complex-valued $\hat{\mathbf{u}}$, then report its absolute value, while ZFDC has to be modified by appending conjugates to $\mathbf{X}$ and $\mathbf{y}$, doubling their size. However, for sequential k-space optimisation, the restriction to real-valued $\mathbf{u}$ (phase contributions are treated as part of the noise $\boldsymbol{\varepsilon}$) is important, keeping the optimisation from wasting

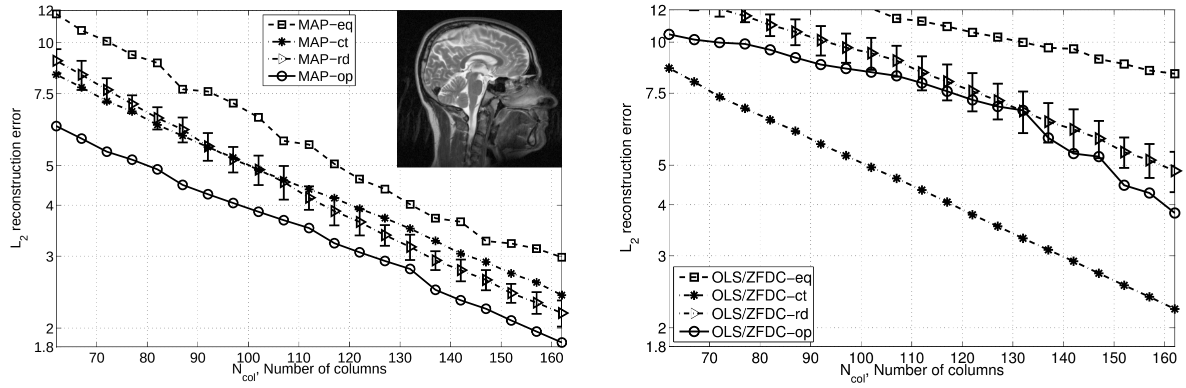


Figure 6.8: Results for Cartesian undersampling, on sagittal slice (TSE, TE=92ms). All designs contain 32 central lines. Equispaced [eq]; low-pass [ct]; random with variable density [rd]; optimised by our Bayesian technique [op], on same slice. Shown are L_2 *distances to* $\mathbf{u}_{true}$*. Left: Nonlinear (MAP) reconstruction. Right: Linear (ZFDC) reconstruction.*

efforts on learning well known symmetries from scratch. For phase-sensitive applications, our method would have to be modified.

6.4.1 Cartesian sequences

In the Cartesian setup, we select individual k-space lines from 256 equispaced candidates (with $d = 256$ samples per line), the complete dataset corresponding to a standard Nyquist-sampled image acquisition. Multiplications with $\mathbf{X}$, $\mathbf{X}_*$ correspond to equispaced discrete Fourier transforms, for which we use the FFTW (Fastest Fourier Transform in the West; `www.fftw.org/`).

All designs compared here start with the 32 lines closest to the origin, which leaves 224 lines to choose from. Based on this low frequency data, we estimate a phase map and post-multiply $\mathbf{X}$ in order to correct for phase noise, as in Lustig et al. [2007]. Phase mapping helps sparse reconstruction, and is vital for Bayesian design optimisation (see Discussion). For the equispaced designs *eq*, the remaining space is covered with $N_{\text{shot}} - 32$ equispaced lines. The low-pass designs *ct* occupy lines densely from the centre outwards. Random designs *rd* are drawn according to the heavy-tailed variable density used in Lustig et al. [2007] (we modified their density to accommodate the smaller central slab), which accounts for the nonuniform spectral distribution of (MR) images specifically ($1/f$ spectral decay). Lines are drawn without replacement. In accordance with Lustig et al. [2007], we noted that drawing lines *uniformly* at random results in poor reconstructions (not shown). Our Bayesian design optimisation technique makes use of the remaining 224 lines as candidate set $\mathcal{C}$. The optimisation is done on a single slice (TSE, TE=92ms, sagittal orientation; figure 6.8, left), featuring many details, while we present test reconstruction results on a wide range of different data, unknown during design optimisation.

Reconstruction error results are given in figure 6.8 (tested on slice used for design optimisation) and figure 6.9 (tested on wide range of other data, unknown during design optimisation). If nonlinear MAP reconstruction is used for undersampled reconstruction, the optimised designs clearly outperform all other choices, especially with fewer lines (the left end, 64 lines, is $\frac{1}{4}$ of the Nyquist rate). Low-pass designs outperform variable density random designs with few lines, while the latter improves from about $\frac{1}{2}$ the Nyquist rate. In contrast, if linear reconstruction is used (figure 6.8, right), only low-pass designs lead to acceptable reconstructions.

Importantly, the dominating part of improvements of optimised over other designs considered here generalises to data never seen during optimisation, as shown in figure 6.9. This is the case even for axial orientations, depicting details different from the single sagittal slice the design was optimised on. As seen in the right panel, the improvements are consistent across echo times, orientations, and subjects, and their size scales with the reconstruction difficulty of the test slice.

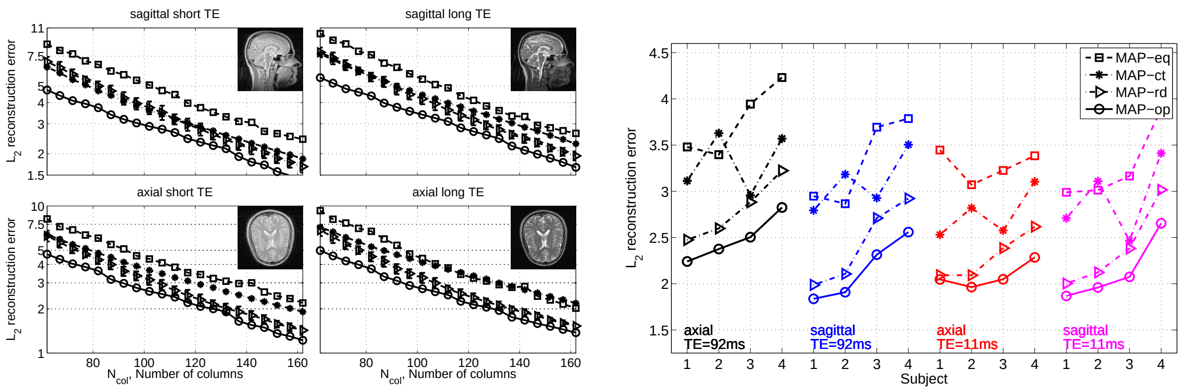


Figure 6.9: Results for Cartesian undersampling, on TSE scans.
The range of data was unknown during design optimisation. We use different echo times (TE=11ms, TE=92ms) and orientations (sagittal, axial). Design choices as in figure 6.8. Shown are L_2 *distances to* $\mathbf{u}_{true}$*, averaged over 5 slices and 4 different subjects. Left: reconstruction test errors for different datasets (echo time, orientation). Error bars for variable density random [rd] w.r.t. ten repetitions. Right: reconstruction test errors, averaged over 5 slices, for designs of 127 lines.*

MAP reconstructions for Cartesian sagittal data (TSE, TE=88ms, unknown during design optimisation) are shown in figure 6.10, for axial data (TSE, TE=92ms) in figure 6.11, comparing different designs of 64 lines ($\frac{1}{4}$ Nyquist; scan time reduction by factor of 4). The superior quality of reconstructions for the optimised design is evident.

6.4.2 Spiral sequences

Interleaved outgoing Archimedian spirals employ k-space trajectories $\mathbf{k}(t) \propto \theta(t)e^{\mathrm{i}2\pi[\theta(t)+\theta_0]}$, $\theta(0) = 0$, where the gradient $\mathbf{g}(t) \propto \mathrm{d}\mathbf{k}/\mathrm{d}t$ grows to maximum strength at the slew rate, then stays there [Bernstein et al., 2004, chapter 17.6]. Sampling along an interleave (azimuthal direction) respects the Nyquist limit. The number of revolutions N_r per interleave, and the number of interleaves N_{shot} determine the radial spacing, with scan time proportional to N_{shot}. We use $N_r = 8$, resulting in 3216 samples per interleave. Radial Nyquist spacing is attained for $N_{\mathrm{shot}} \geq 16$. Candidates are interleaves, parametrised by the offset angle: $\mathbf{X}_* = \mathbf{X}_*(\theta_0)$, with $d = 3216$ rows. Samples do not lie on a regular grid: non-equispaced FFT is used to multiply with $\mathbf{X}$, $\mathbf{X}_*$ (NFFT with Kaiser-Bessel kernel [Bernstein et al., 2004, chapter 13.2]; `www-user.tu-chemnitz.de/~potts/nfft`). Our experiments are idealised, in that spiral sampling is simulated by NFFT interpolation from data acquired on a grid.

We compare MAP reconstruction under a number of design choices: equispaced (*eq*), uniformly drawn at random (*rd*), and optimised (*op*). Angles lie in $[0, 2\pi)$ in the first, and in $[0, \pi)$ in the second setting. All designs contain $\theta_0 = 0$. In addition, *eq* uses $\theta_0 = j(k\pi/N_{\mathrm{shot}})$, $j = 1, \dots, N_{\mathrm{shot}} - 1$; for *rd*, we draw $N_{\mathrm{shot}} - 1$ angles uniformly at random from $\mathcal{C} = (k\pi/256)[1 : 255]$, averaging results over ten repetitions; for *op*, we start from the single interleave $\theta_0 = 0$ and use the candidate set $\mathcal{C}$. Here, $k \in \{1, 2\}$, depending on the setting. For $k = 2$, setups with $N_{\mathrm{shot}} = 8$ halve the scan time, compared to Nyquist spacing. Designs are optimised on a single slice (figure 6.8, left), featuring many details.

In the first setting ($k = 2$), the near-Hermitian symmetry of data means that *eq* is at a disadvantage for even N_{shot}. In order to correct for this fact, and to test the relevance of $\mathbf{u}$ being close to real-valued (after phase mapping and subtraction), we restrict angles to $[0, \pi)$ in a second setting ($k = 1$). By interpolating non-Cartesian sampling, we ignore characteristic errors of spiral acquisition in practise, which may diminish the impact of our findings (see section 6.5).

MAP reconstruction errors for spiral undersampling are given in figure 6.12. The left column shows performance on the data the angles were optimised over, while in the right column,

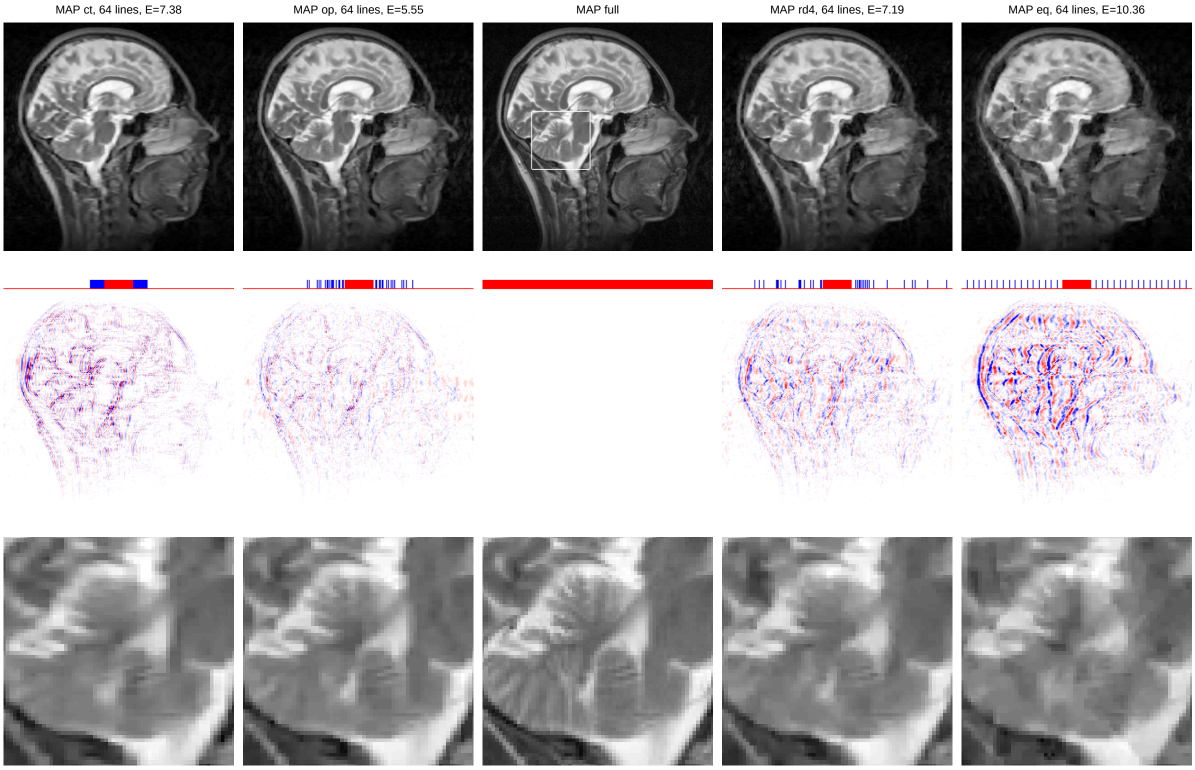


Figure 6.10: MAP reconstructions for Cartesian undersampling, sagittal TSE data. We have TE=88ms (unknown during design optimisation) and $N_{shot} = 64$ phase encodes (red: 32 initial centre lines; blue: 32 additional encodes according to design choices). Upper row: full images. White window: location of blow-up. Middle row: residuals (difference to $\mathbf{u}_{true}$), location of phase encodes (k-space columns). Lower row: blow-ups.
MAP ct: apparent lower resolution, fine structures smoothed out. MAP rd: erroneous dark structure (upper left). MAP op: satisfying level of details at $\frac{1}{4}$ of Nyquist rate, considerably more detail and less blurring than for the other undersampled designs.

we test generalisation behaviour on a range of different data. The lower row corresponds to the first setting, with offset angles $\theta_0 \in [0, 2\pi)$. As expected, *eq* for even N_{shot} does poorly, due to the almost-Hermitian symmetry of the data, while performing comparably to *op* for odd N_{shot}. In the second setting ($\theta_0 \in [0, \pi)$, upper row), *eq* and *op* perform similarly from $N_{\text{shot}} = 7$, with *op* outperforming *eq* for smaller designs. In comparison, drawing offset angles at random leads to much worse MAP reconstructions in either setting. As for Cartesian undersampling, the performance on different datasets, unknown at optimisation time, is comparable to the behaviour on the training set, except that *eq* does substantially worse on axial than on sagittal scans.

6.5 Discussion

We have highlighted the importance of k-space sampling optimisation tailored specifically to novel nonlinear sparse reconstruction algorithms, and have proposed a Bayesian experimental design framework. Our experimental results for Cartesian undersampling show that sparse reconstruction quality depends strongly on the sampling design chosen, with phase encodes optimised by our Bayesian technique outperforming other commonly used undersampling schemes, such as low-pass or variable density random designs [Lustig et al., 2007]. With optimised sampling, high-quality reconstructions are obtained if only half of all lines are measured, and useful images can be reconstructed at $\frac{1}{4}$ of the Nyquist rate (figure 6.10, figure 6.11). The behaviour of undersampling designs is very different for linear reconstruction, where only low-pass measurements lead to good reconstructions (figure 6.8, right), indicating that linear design optimisation concepts, such as the point spread function (see section 6.2.3), play a diminished

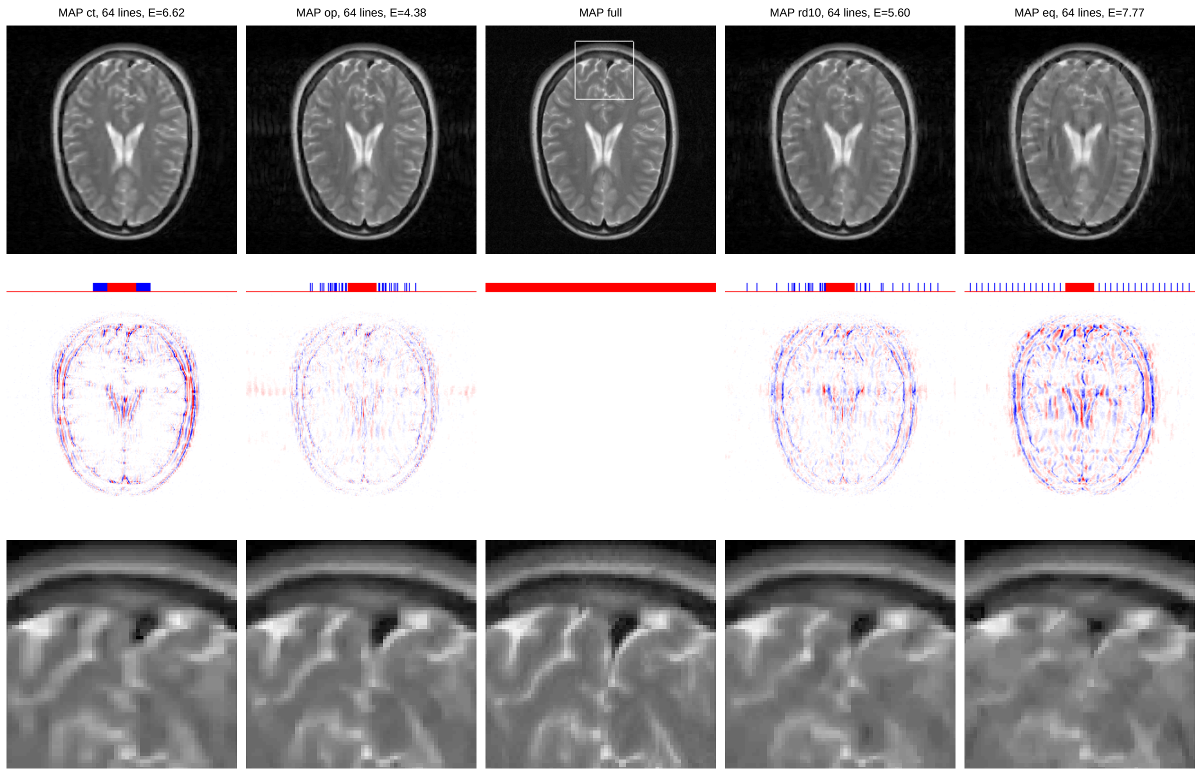


Figure 6.11: MAP reconstructions for Cartesian undersampling, axial TSE data. We have TE=11ms (unknown during design optimisation) and $N_{shot} = 64$ phase encodes (red: 32 initial centre lines; blue: 32 additional encodes according to design choices). Upper row: full images. White window: location of blow-up. Middle row: residuals (difference to $\mathbf{u}_{true}$), location of phase encodes (k-space columns). Lower row: blow-ups. MAP ct: apparent lower resolution than MAP rd, MAP op. Both MAP ct and MAP rd have tendency to fill in dark area. MAP op retains high contrast there.

role for nonlinear reconstruction, and that sampling optimisation has to be matched to the reconstruction modality. The improvement of optimised over other design choices is most pronounced for fewer number of lines acquired. Importantly, even though designs are optimised on a single slice of data, a large part of these improvements generalises to different datasets in our study, featuring other slice positions, subjects, echo times, and even orientations (figure 6.9). Our results indicate that Bayesian design optimisation can be used offline, adjusting trajectories on data acquired under controlled circumstances, and final optimised designs can be used for future scans. Our framework embodies the idea of adaptive optimisation. The sampling design is adjusted based on a representative dataset (called training set), and if adequate measures for complexity control are in place (Bayesian sparsity prior, proper representation of posterior mass, sequential scheme of uncovering information only if asked for), good performance on the training set (figure 6.8) tends to imply good performance on independent test sets (figure 6.9), thus successful generalisation to similar future tasks.

Our framework is not limited to Cartesian sampling, as demonstrated by our application to spiral *k*-space optimisation. However, our findings are preliminary in this case: spiral sampling was interpolated from data acquired on a Cartesian grid, and only the offset angles of dense Archimedian interleaves were optimised (instead of considering variable-density spiral interleaves as well). In this setting, designs optimised by our technique show comparable performance to spacing offset angles equally, while a randomisation of these angles performs much worse.

In Bayesian design optimisation, statistical information is extracted from one or few representative images used during training and represented in the posterior distribution, which serves as oracle to steer further acquisitions along informative directions. Importantly, and con-

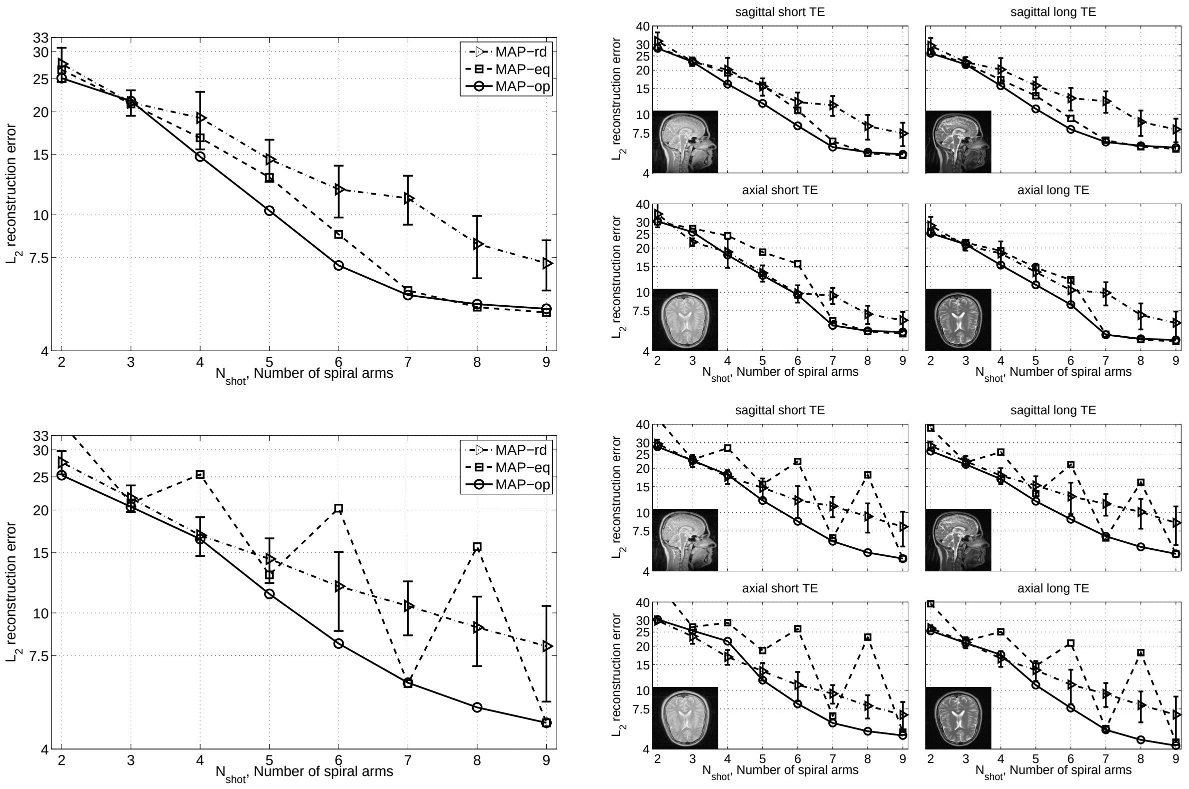


Figure 6.12: Results for MAP reconstruction, spiral undersampling of offset angles θ_0. Left column: reconstruction errors on sagittal slice (see figure 6.8 left), on which op *is optimised. Right column: reconstruction errors on different data (averaged over 5 slices, 4 subjects each, see figure 6.9). Upper row: offset angles from $[0, \pi)$. Lower row: offset angles from $[0, 2\pi)$. Design choices: equispaced [eq]; uniform at random [rd] (averaged over 10 repetitions); optimised by our Bayesian technique [op].*

firmed in further experiments (not shown), it is essential to optimise the design on MRI data for real-world subjects, or controlled objects of similar statistical complexity; simple phantoms do not suffice. While the latter are useful to analyse linear reconstruction, they cannot play the same role for nonlinear sparse reconstruction. Modern theory proves that overly simple signals (such as piecewise constant phantoms) are reconstructed perfectly from undersampled measurements, almost independently of the design used for acquisition [Candès et al., 2006, Donoho, 2006a]. This advantage of sparse reconstruction *per se*, for almost any design, does not carry over to real-world images such as photographs (see chapter 5) or clinical resolution MR images. The relevance of design optimisation grows with the signal complexity, and is dominatingly present for MR images of diagnostically useful content and resolution.

Variable density phase encoding sampling does not perform well at $\frac{1}{4}$ of the Nyquist rate (figure 6.10, figure 6.11), if the density of Lustig et al. [2007] is used. For a different density with lighter tails (more concentrated on low frequencies), reconstructions are better at that rate, but are significantly worse at rates approaching $\frac{1}{2}$ or more (results not shown). In practise, this drawback can be alleviated by modifying the density as the number of encodes grows. From our experience, a second major problem with variable density design sampling comes from the independent nature of the process: the inherent variability of independent sampling leads to uncontrolled gaps in *k*-space, which tend to hurt image reconstruction substantially. Neither of these problematic aspects is highlighted in Lustig et al. [2007], or in much of the recent compressed sensing theory, where incoherence of a design is solely focused on. A clear outcome from our experiments here is that while incoherence plays a role for nonlinear reconstruction, its benefits are easily outweighed by neglecting other design properties. Once design sampling distributions have to be modified with the number of encodes, and dependencies to previously drawn encodes have to be observed, the problem of choosing such a scheme is equivalent to the design optimisation problem, for which we propose a data-driven alternative to trial-and-error here, showing how to partly automatise a laborious process, which in general has to be repeated from scratch for every new configuration of scanner setup and available signal prior knowledge.

Further issues will have to be addressed in a fully practical application of our method. We extracted (or interpolated) undersampling trajectories from data acquired on a complete Cartesian grid, which may be realistic for Cartesian undersampling, but neglects practical inaccuracies specific to non-Cartesian trajectories. Moreover, in multi-echo sequences, the ordering of phase encodes matters. For an immobile training subject/object, our sequential method can be implemented by nested acquisitions: running a novel (partial) scan whenever $\mathbf{X}$ is extended by a new interleave, dropping the data acquired previously. With further attendance to implementation and commodity hardware parallelisation, the time between these scans will be on the order of a minute. Gradient and transmit or receive coil imperfections (or sensitivities), as well as distortions from eddy currents, may imply constraints for the design, so that less candidates may be available in each round. Such adjustments to reality will be simplified by the inherent configurability of our Bayesian method, where likelihood and prior encode forward model and known signal properties.

The near-Hermitian symmetry of measurements is an important instance of prior knowledge, incorporated into our technique by placing sparsity potentials on the imaginary part $\Im(\mathbf{u})$. This leads to marked improvements for sparse reconstruction, and is essential for Bayesian *k*-space optimisation to work well. In addition, phase mapping and subtraction is required. Phase contributions substantially weaken image sparsity statistics, thereby eroding the basis sparse reconstruction stands upon. In the presence of unusual phase errors, specialised phase mapping techniques should be used instead. In future work, we aim to integrate phase mapping into our framework.

In light of the absence of a conclusive nonlinear *k*-space sampling theory and the well-known complexity of nonlinear optimal design, our approach has to be seen in the context of other realizable strategies. Designs can optimised by blind (or heuristic) trial-and-error exploration [Marseille et al., 1996], which in general is much more demanding in terms of human expert and MRI scan time than our approach. Well-founded approaches fall in two classes: artificially simplified problems are solved optimally, or adaptive optimisation on representative *real* datasets is used. We have commented above on recent advances in the first class, for extremely sparse, unstructured signals [Candès et al., 2006, Donoho, 2006a], but these results empirically seem to carry little relevance for real-world signals. Our method falls in the second class, as an instance of nonlinear sequential experimental design [Chaloner and Verdinelli, 1995, Fedorov, 1972], where real-world signals are addressed directly, and for which few practically relevant performance guarantees are available. Our approach to design optimisation is sequential, adapting measurements to largest remaining uncertainties in the reconstruction of a single image. While we established sound generalisation behaviour on unseen data in our experiments, real-time MRI [Gamper et al., 2008], [Bernstein et al., 2004, chapter 11.4] may especially benefit from our sequential, signal-focused approach. While our algorithm at present does not attain the high frame rates required in these applications, algorithmic simplifications, combined with massively parallel digital computation, could allow our framework to be used in the future in order to provide advanced data analysis and decision support to an operator during a running MRI diagnosis.

Possible extensions include the application of the framework to 3D imaging. One step in this direction has already been done by Seeger [2010b], where Markovian assumptions between neighbouring slices are used to approximate full inference on a 3D cube of voxels instead of a 2D slice. Other future steps include the application of our methodology to real non-Cartesian measurements instead of simulated ones.

Chapter 7

Overall Conclusion and Perspectives

7.1 Summary

In this thesis, we developped and discussed many aspects of deterministic approximate inference algorithms for generalised linear Bayesian models: chapter 3 focused on convexity and scalability, chapter 4 compared relative accuracy. We applied the algorithms to binary classification (chapter 3), linear compressive image acquisition (chapter 5) and magnetic resonance imaging (MRI) optimisation (chapter 6) proving the validity and utility of our approach.

We studied three kinds of problems in increasing order of difficulty:

1. estimation, where the probabilistic model needs to provide a single best *answer*, that means a decision used in the future,
2. inference, where a normalised relative weighting between *all possible answers* in form of the posterior distribution is provided leaving the decision open, and
3. experimental design, where we seek to determine the *questions* to be asked in the first place to obtain solid knowledge allowing to produce informed answers subsequently.

In order to overcome analytical intractabilities, we had to do several approximations: we replaced non-Gaussian distributions by Gaussian ones and we worked with lower bounds on marginal variances instead of their exact values. We saw strong similarities between the approximate inference algorithms allowing to understand the effect of the approximations in practise. Also, we made clear that inference is to a certain extent orthogonal to modelling because many inference algorithms are able to approximate the exact posterior using the same interface. We also detailed the nested structure of the interrelations between estimation, inference and design: design can be done using a sequence of inference steps and inference can be understood as a sequence of estimation steps. Most estimators are solutions of *optimisation problems*; on the contrary, inference corresponds to considerably harder *integration problems*.

7.2 Discussion and Outlook

We group the ideas on possible extensions of the work and future research directions into three different categories: theory, algorithms and applications.

Theory

The focus of this thesis is more on computations than on pure analysis. Therefore, some theoretical questions do remain. In continuous optimisation (or equivalently estimation techniques), it is convex problems (log-concave unimodal models) not linear ones (Gaussian models) that are considered simple [Boyd and Vandenberghe, 2004]. Similarly, we were able to show that there is a similar line of separation in a particular approach to inference: our variational algorithm is scalable, convex and convergent for log-concave and not only for Gaussian models. However it is unclear, how general this statement is.

Furthermore, a Gaussian approximation captures pairwise interactions but higher-order dependencies remain impossible to be represented. For large image models, already relationships between every pair of pixels are very challenging. Also, it would be interesting to formalise, how much of the non-Gaussian behaviour such as sparsity can be conserved in principle by a sequence of Gaussians as we use it.

The relationship between inference and estimation is not yet fully understood in general, especially for high-dimensional and non-Gaussian models. Inference is certainly computationally harder but also offers some benefits. Sometimes inference problems have less local minima than the corresponding estimation problems [Wipf et al., 2010]. The result of an inference procedure provides an intrinsic sensitivity statement.

Finally, linear experiment design is widely used in biology for example. Non-linear and/or Bayesian experimental design has received much less attention in the statistics literature even though it can deal much better with the underdetermined case. There is surprisingly little theoretical analysis for sequential non-Gaussian design.

Algorithms

The most important algorithmical questions concern convexity and scalability. Are there versions of Gaussian Kullback-Leibler divergence minimisation (KL) or even expectation propagation (EP) that can be solved by convex optimisation? What is so special about the combination of the variational Bayesian (VB) relaxation and the decoupling idea so that it yields a scalable algorithm? It would be very interesting to further analyse whether there is in fact a trade-off between accuracy (VB is less accurate than EP) and scalability (EP is not scalable), or whether there is a way of deriving a scalable EP or KL algorithm. Also, one can try and improve the variance lower bounds we used.

A more obvious step would not alter the algorithms but would rather improve the implementation. Modern parallel processors and graphics cards offer a lot of computing power able to alleviate the computational burden substantially.

Applications

Generalised and sparse linear models are omnipresent cornerstones of applied statistics heavily employed in information retrieval, machine learning, computational biology, signal processing and econometrics. Our inference technology is valuable whenever there is the need to not only output a single decision but accomplish a higher-level task: optimisation of the linear map itself according to information theoretic criteria. If the space of linear maps (e.g. the image measurement architecture) has many parameters, it is impossible to sample by a human expert. Here, our design algorithms can help the exploration process by simulating parts on a computer reducing the number of necessary real-world experiments.

Undersampling or more generally exploiting redundancy in signals to accelerate their acquisition is only a particular instance of the trend where more computational power in a post-processing step can compensate for an incomplete or noisy acquisition step. Our methodology allows to optimise the acquisition step in this scenario.

A particularly interesting domain is image processing, where linear and bilinear models are used, e.g. for removing camera shake [Fergus et al., 2006] using inference techniques [Miskin, 2000]. Our variational framework can be applied here, as well.

Finally, our MRI imaging study was only a first step, many more are possible. Experiment design to speed up the acquisition of three-dimensional spatial volumes, four-dimensional spatio-temporal data possibly using parallel receiver coils is and remains challenging.

Appendix A

Matrix and Differential Calculus

A.1 Inverses, determinants and generalised inverses

A.1.1 Matrix inversion lemma

The numerical inversion of a non-singular matrix $\mathbf{A} \in \mathbb{R}^{n \times n}$ is an $\mathcal{O}(n^3)$ operation. From $\mathbf{A}^{-1}$, one can compute the inverse of a rank-k modified matrix $\mathbf{A} + \mathbf{UBV}^\top$ in $\mathcal{O}(k \cdot n^2)$ by the so-called *Sherman–Morrison–Woodbury formula* or simply the *Woodbury formula* [Woodbury, 1950]. Precisely, for invertible $\mathbf{B} \in \mathbb{R}^{k \times k}$ and general $\mathbf{U}, \mathbf{V} \in \mathbb{R}^{n \times k}$ we have

$$\left(\mathbf{A} + \mathbf{UBV}^\top\right)^{-1} = \mathbf{A}^{-1} - \mathbf{A}^{-1}\mathbf{U}\left(\mathbf{B}^{-1} + \mathbf{V}^\top\mathbf{A}^{-1}\mathbf{U}\right)^{-1}\mathbf{V}^\top\mathbf{A}^{-1},$$

which simplifies for $k = 1$, $\mathbf{B} = 1$, $\mathbf{U} = \mathbf{u}$, $\mathbf{V} = \mathbf{v}$ to the *Sherman-Morrison identity*

$$\left(\mathbf{A} + \mathbf{uv}^\top\right)^{-1} = \mathbf{A}^{-1} - z \cdot \mathbf{xy}^\top,\ \mathbf{x} = \mathbf{A}^{-1}\mathbf{u},\ \mathbf{y} = \mathbf{A}^{-1}\mathbf{v},\ z = \frac{1}{1 + \mathbf{v}^\top\mathbf{A}^{-1}\mathbf{u}}$$

for rank-1 updating an matrix inverse.

A.1.2 Matrix determinant lemma

A similar identity exists for the update of a determinant of a matrix under the name *general matrix determinant lemma*

$$\left|\mathbf{A} + \mathbf{UBV}^\top\right| = \left|\mathbf{B}^{-1} + \mathbf{V}^\top\mathbf{A}^{-1}\mathbf{U}\right| \cdot |\mathbf{B}| \cdot |\mathbf{A}|,$$

which includes the *matrix determinant lemma* as the special case $k = 1$, $\mathbf{B} = 1$, $\mathbf{U} = \mathbf{u}$, $\mathbf{V} = \mathbf{v}$

$$\left|\mathbf{A} + \mathbf{uv}^\top\right| = (1 + \mathbf{v}^\top\mathbf{A}^{-1}\mathbf{u}) \cdot |\mathbf{A}|.$$

If $\mathbf{A}^{-1}$ is known already, the determinant can be updated in $\mathcal{O}(k \cdot n^2)$ as well.

A.1.3 Generalised inverses and pseudoinverse

For a non-singular quadratic matrix $\mathbf{A}$, the *matrix inverse* is the unique matrix $\mathbf{B}$ satisfying $\mathbf{AB} = \mathbf{BA} = \mathbf{I}$ denoted by $\mathbf{A}^{-1}$.

While loosing some of the properties of a proper matrix inverse, the concept can be generalised to singular and rectangular matrices. A *generalised inverse* $\mathbf{A}^- \in \mathbb{R}^{n \times m}$ of matrix $\mathbf{A} \in \mathbb{R}^{m \times n}$ [Lütkepohl, 1997] has to satisfy

$$\mathbf{AA}^-\mathbf{A} = \mathbf{A}.$$

This construction is not unique, since for any matrix $\mathbf{B} \in \mathbb{R}^{n \times m}$, the matrix $\tilde{\mathbf{A}}^- := \mathbf{A}^- + \mathbf{B} - \mathbf{A}^-\mathbf{ABAA}^-$ is also a generalised inverse of $\mathbf{A}$. Only for $m = n$ and non-singular $\mathbf{A}$, the generalised inverse and the inverse are the same $\mathbf{A}^- = \mathbf{A}^{-1}$. Examples include the *Drazin inverse* for singular quadratic matrices and the *Bott-Duffin inverse* from constrained optimisation for rectangular matrices.

By far the most prominent generalised inverse is the unique *Moore-Penrose pseudo inverse* $\mathbf{A}^+$ obeying

$$\mathbf{A}^+\mathbf{A}\mathbf{A}^+ = \mathbf{A}^+, \quad \mathbf{A}\mathbf{A}^+ = \left(\mathbf{A}\mathbf{A}^+\right)^\top, \quad \mathbf{A}^+\mathbf{A} = \left(\mathbf{A}^+\mathbf{A}\right)^\top$$

in addition. It can be computed from the compact singular value decomposition $\mathbf{A} = \mathbf{U}\mathbf{S}\mathbf{V}^\top$ with orthonormal $\mathbf{U}, \mathbf{V} \in \mathbb{R}^{n\times r}$, diagonal $\mathbf{S} \in \mathbb{R}_+^{r\times r}$ and r being the rank of $\mathbf{A}$ by

$$\mathbf{A}^+ = \mathbf{U}\mathbf{S}^{-1}\mathbf{V}^\top.$$

Another way of obtaining $\mathbf{A}^+$ is based on the limit $\mathbf{A}^+ = \lim_{\delta\to 0} \mathbf{A}^\top(\mathbf{A}\mathbf{A}^\top + \delta\mathbf{I})^{-1}$ and equivalently on $\mathbf{A}^+ = \lim_{\delta\to 0}(\mathbf{A}^\top\mathbf{A} + \delta\mathbf{I})^{-1}\mathbf{A}^\top$.

A.2 Derivatives and differential calculus

For a function $f : \mathbb{R}^n \to \mathbb{R}^m$ and a point $\mathbf{a} \in \mathbb{R}^n$, we call the unique linear function $\lambda : \mathbb{R}^n \to \mathbb{R}^m$ satisfying

$$\lim_{\mathbf{h}\to 0} \frac{\|f(\mathbf{a}+\mathbf{h}) - f(\mathbf{a}) - \lambda(\mathbf{h})\|}{\|\mathbf{h}\|} = 0 \tag{A.1}$$

the *derivative* of f at $\mathbf{a}$. We use the concept of Fréchet derivative in the following since it is most adapted to be used in the matrix calculus. The function f comes from a space $\mathcal{F}$ and the subspace of $\mathcal{F}$ containing linear functions only is denoted by $\mathcal{L}$. Equation A.1 formalises the notion that λ is an optimal local linear approximation to f at $\mathbf{a}$. Every linear function, $\lambda : \mathbb{R}^n \to \mathbb{R}^m$ can be represented by an $m \times n$ matrix $\mathbf{G}$ so that $\lambda(\mathbf{x}) = \mathbf{G}\mathbf{x}$. Since all information about λ is contained in the matrix $\mathbf{G}$, we often talk about the matrix $\mathbf{G} \in \mathbb{R}^{m\times n}$ when we actually reason about the function $\lambda \in \mathcal{L}$. We use the notation $\mathrm{d}f(\mathbf{a}) : \mathbb{R}^n \to \mathbb{R}^m \in \mathcal{L}$ to refer to the derivative of $f \in \mathcal{F}$ at $\mathbf{a}$ (i.e. the $\lambda \in \mathcal{L}$ satisfying condition A.1) and the *differential* $\mathrm{d}f : \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}^m$ is employed whenever we want to work with a generic value of $\mathbf{a}$. For the case of scalar outputs, i.e. $m = 1$, the matrix $\mathbf{G} \in \mathbb{R}^{1\times n}$ specifying the behaviour of $\mathrm{d}f(\mathbf{a})$ is denoted by $f'(a)$, $\frac{\partial f}{\partial \mathbf{x}}(\mathbf{a})$ or $\frac{\partial f}{\partial \mathbf{X}}(\mathbf{A})$ depending on the input. Finally, we use the notation

$$\begin{array}{rll}
\text{differential} & \text{derivative} & \\
\mathrm{d}f = \mathrm{d}x \cdot f' & \mathrm{d}f(x) = \mathrm{d}x \cdot f'(x) & \text{(A.2)} \\
\mathrm{d}f = \mathrm{d}\mathbf{x}^\top \dfrac{\partial f}{\partial \mathbf{x}} & \mathrm{d}f(\mathbf{x}) = \mathrm{d}\mathbf{x}^\top \dfrac{\partial f}{\partial \mathbf{x}}(\mathbf{x}) & \\
\mathrm{d}f = \mathrm{tr}\left(\mathrm{d}\mathbf{X}^\top \dfrac{\partial f}{\partial \mathbf{X}}\right) & \mathrm{d}f(\mathbf{X}) = \mathrm{tr}\left(\mathrm{d}\mathbf{X}^\top \dfrac{\partial f}{\partial \mathbf{X}}(\mathbf{X})\right) & \text{(A.3)}
\end{array}$$

for the differentials and the derivatives, i.e. the linear mappings $\mathrm{d}\mathbf{x} \mapsto \mathbf{z} \in \mathbb{R}^m$, where $\mathbf{z}$ equals $\mathrm{d}f(\mathbf{x})$ "evaluated at" the small change $\mathrm{d}\mathbf{x}$. The reason why the above notation is so powerful comes from the fact that it encompasses derivatives of vector and matrix valued functions in a common framework using the standard calculus from linear algebra avoiding nasty summations and multi-indices. For a good reference, see Magnus and Neudecker [1999].

For gradient-based optimisation, one is often interested in deriving an expression for the vector $\frac{\partial f}{\partial \mathbf{x}}(\mathbf{x})$. In order to do that, there are some rules that allow – starting from $\mathrm{d}f$ – to obtain expressions of the form of equation A.3, where one can simply read off $\frac{\partial f}{\partial \mathbf{x}}(\mathbf{x})$. In the following, we list some handy rules to manipulate differential expressions [Lütkepohl, 1997].

A.2.1 Simple rules

Among the simple rules, we have $\mathrm{d}\mathbf{A} = \mathbf{0}$ for constant expressions as well as $\mathrm{d}(a\mathbf{X} + b\mathbf{Y}) = a\mathrm{d}\mathbf{X} + b\mathrm{d}\mathbf{Y}$, $\mathrm{dtr}(\mathbf{X}) = \mathrm{tr}(\mathrm{d}\mathbf{X})$, $\mathrm{d}(\mathrm{diag}(\mathbf{X})) = \mathrm{diag}(\mathrm{d}\mathbf{X})$ and $\mathrm{d}(\mathbf{X}^\top) = (\mathrm{d}\mathbf{X})^\top$ for linear expressions.

A.2.2 Product rules

For matrix products and Hadamard products, we have the rules $\mathrm{d}(\mathbf{XY}) = \mathrm{d}\mathbf{XY} + \mathbf{X}\mathrm{d}\mathbf{Y}$ and $\mathrm{d}(\mathbf{X} \odot \mathbf{Y}) = \mathrm{d}\mathbf{X} \odot \mathbf{Y} + \mathbf{X} \odot \mathrm{d}\mathbf{Y}$ implying $\mathrm{d}(\mathbf{X}^n) = \sum_{i=1}^{n} \mathbf{X}^{i-1}\mathrm{d}\mathbf{X}\mathbf{X}^{n-i}$.

A.2.3 Determinant, inverse and pseudo-inverse

In the following, we list $\mathrm{d}f(\mathbf{X})$ for some common matrix valued functions.

$$\begin{aligned}
\mathrm{d}|\mathbf{X}| &= |\mathbf{X}| \cdot \mathrm{tr}(\mathbf{X}^{-1}\mathrm{d}\mathbf{X}) \\
\mathrm{d}\ln|\mathbf{X}| &= \mathrm{tr}(\mathbf{X}^{-1}\mathrm{d}\mathbf{X}) \\
\mathrm{d}\mathbf{X}^{-1} &= -\mathbf{X}^{-1}\mathrm{d}\mathbf{X}\mathbf{X}^{-1}
\end{aligned}$$

The pseudo inverse does not admit a closed-form expression for $\mathrm{d}f(\mathbf{X})$, however, we can write:

$$\mathbf{X}\mathrm{d}\mathbf{X}^{+}\mathbf{X} = -\mathbf{X}\mathbf{X}^{+}\mathrm{d}\mathbf{X}\mathbf{X}^{+}\mathbf{X}.$$

A.2.4 Matrix exponential

The matrix valued function defined by $e^{\mathbf{X}} = \sum_{k=0}^{\infty} \frac{1}{k!}\mathbf{X}^k$, called the matrix exponential, is distinctively different from the component-wise matrix exponentiation $[\exp(\mathbf{X})]_{ij} = \exp(X_{ij})$.

$$\begin{aligned}
\mathrm{d}\exp(\mathbf{X}) &= \exp(\mathbf{X}) \odot \mathrm{d}\mathbf{X} \\
\mathrm{dtr}(e^{\mathbf{X}}) &= \mathrm{tr}(e^{\mathbf{X}}\mathrm{d}\mathbf{X})
\end{aligned}$$

A.2.5 Matrix decompositions

Singular values for general $\mathbf{X} \in \mathbb{R}^{m \times n}$:

$$\begin{aligned}
\mathbf{X} &= \mathbf{U\Sigma V}^\top,\ \mathbf{U}^\top\mathbf{U} = \mathbf{I},\ \mathbf{\Sigma} = \mathrm{dg}(\sigma),\ \mathbf{V}^\top\mathbf{V} = \mathbf{I} \\
\mathrm{d}\sigma &= \mathrm{dg}(\mathbf{U}^\top\mathrm{d}\mathbf{X}\mathbf{V})
\end{aligned}$$

Eigenvalues for symmetric $\mathbf{X} \in \mathrm{Sym}_n = \{\mathbf{X} \in \mathbb{R}^{n \times n} : \mathbf{X} = \mathbf{X}^\top\}$:

$$\begin{aligned}
\mathbf{X} &= \mathbf{V\Lambda V}^\top,\ \mathbf{V}^\top\mathbf{V} = \mathbf{I},\ \mathbf{\Sigma} = \mathrm{dg}(\boldsymbol{\lambda}) \\
\mathrm{d}\boldsymbol{\lambda} &= \mathrm{dg}(\mathbf{V}^\top\mathrm{d}\mathbf{X}\mathbf{V})
\end{aligned}$$

Eigenvectors for symmetric $\mathbf{X} \in \mathrm{Sym}_n$ $\forall i = 1..n$:

$$\begin{aligned}
\mathbf{X}\mathbf{v}_i &= \lambda_i\mathbf{v}_i,\ \mathbf{v}_i^\top\mathbf{v}_i = 1 \\
\mathrm{d}\mathbf{v}_i &= (\lambda_i\mathbf{I} - \mathbf{X})^{+}\mathrm{d}\mathbf{X}\mathbf{v}_i = -\sum_{j=1,j\neq i}^{n} \mathbf{v}_j \frac{1}{\lambda_j - \lambda_i}\mathbf{v}_j^\top\mathrm{d}\mathbf{X}\mathbf{v}_i
\end{aligned}$$

A.2.6 General spectral functions

The section is based on Lewis [1996]. For $\mathbf{X} \in \mathrm{Sym}_n$, a spectral function $\phi : \mathrm{Sym}_n \to \mathbb{R}$ satisfies $\phi(\mathbf{UXU}^\top) = \phi(\mathbf{X})$ for any orthonormal matrix $\mathbf{U} \in \mathrm{SO}_n = \{\mathbf{X} \in \mathbb{R}^{n \times n} : \mathbf{U}^\top\mathbf{U} = \mathbf{UU}^\top = \mathbf{I}\}$. Denoting by $\boldsymbol{\lambda} : \mathrm{Sym}_n \to \mathbb{R}^n$ or $\mathbf{\Lambda} : \mathrm{Sym}_n \to \mathbb{R}^{n \times n}$ the eigenvalue function $\mathbf{X} \mapsto [\lambda_1(\mathbf{X}), .., \lambda_n(\mathbf{X})]^\top$ or $\mathbf{X} \mapsto \mathrm{dg}[\lambda_1(\mathbf{X}), .., \lambda_n(\mathbf{X})]$ returning the vector with the ordered eigenvalues $\lambda_1 \geq \lambda_2 \geq .. \geq \lambda_n$, every spectral function can be written as $\phi(\mathbf{X}) = f(\boldsymbol{\lambda}(\mathbf{X})) = f(\mathbf{\Lambda}(\mathbf{X}))$ for a symmetric function $f : \mathbb{R}^n \mapsto \mathbb{R}$. Hence, the name spectral function; $\phi(\mathbf{X})$ only depends on the spectrum $\boldsymbol{\lambda}(\mathbf{X})$. The differential is then given by

$$\mathrm{d}\phi(\mathbf{X}) = \mathrm{tr}\left((f \circ \boldsymbol{\lambda})'(\mathbf{X})\mathrm{d}\mathbf{X}\right) = \mathrm{tr}\left(\mathbf{U}f'(\mathbf{\Lambda})\mathbf{U}^\top\mathrm{d}\mathbf{X}\right),\ \text{where } \mathbf{X} = \mathbf{U\Lambda U}^\top.$$

Two special cases are interesting:

- For $f(\boldsymbol{\lambda}) = \mathbf{1}^\top \ln \boldsymbol{\lambda}$, $\phi(\mathbf{X}) = \ln|\mathbf{X}|$ and $f'(\boldsymbol{\lambda}) = \boldsymbol{\lambda}^{-1}$, we obtain $\mathrm{d}\ln|\mathbf{X}| = \mathrm{tr}\left(\mathbf{U}\boldsymbol{\Lambda}^{-1}\mathbf{U}^\top\mathrm{d}\mathbf{X}\right) = \mathrm{tr}\left(\mathbf{X}^{-1}\mathrm{d}\mathbf{X}\right)$.
- For $f(\boldsymbol{\lambda}) = \mathbf{1}^\top e^{\boldsymbol{\lambda}}$, $\phi(\mathbf{X}) = \mathrm{tr}(e^{\mathbf{X}})$ and $f'(\boldsymbol{\lambda}) = e^{\boldsymbol{\lambda}}$, we obtain $\mathrm{d}\mathrm{tr}(e^{\mathbf{X}}) = \mathrm{tr}\left(\mathbf{U}e^{\boldsymbol{\Lambda}}\mathbf{U}^\top\mathrm{d}\mathbf{X}\right) = \mathrm{tr}\left(e^{\mathbf{X}}\mathrm{d}\mathbf{X}\right)$.

More generally, the differential of the matrix valued function $F : \mathrm{Sym}_n \to \mathrm{Sym}_n$ obeying $F(\mathbf{X}) = \mathbf{U}f(\boldsymbol{\Lambda})\mathbf{U}^\top$ with $\mathbf{X} = \mathbf{U}\boldsymbol{\Lambda}\mathbf{U}^\top$ is harder to obtain

$$\begin{aligned}
\mathrm{d}F(\mathbf{X}) &= \mathbf{U}f'(\boldsymbol{\Lambda})\mathrm{d}\boldsymbol{\Lambda}\mathbf{U}^\top + \mathrm{d}\mathbf{U}f(\boldsymbol{\Lambda})\mathbf{U}^\top + \mathbf{U}f(\boldsymbol{\Lambda})\mathrm{d}\mathbf{U}^\top \\
&= \mathbf{U}\mathrm{dg}\left[f'(\boldsymbol{\lambda}) \odot \mathrm{dg}(\mathbf{U}^\top\mathrm{d}\mathbf{X}\mathbf{U})\right]\mathbf{U}^\top + \sum_{i=1}^{n} \mathbf{u}_i f_i(\lambda_i)\mathrm{d}\mathbf{u}_i^\top + \mathrm{d}\mathbf{u}_i f_i(\lambda_i)\mathbf{u}_i^\top \\
&= \sum_{i=1}^{n} \mathbf{u}_i f_i'(\lambda_i)\mathbf{u}_i^\top\mathrm{d}\mathbf{X}\mathbf{u}_i\mathbf{u}_i^\top + \mathbf{u}_i f_i(\lambda_i)\mathrm{d}\mathbf{u}_i^\top + \mathrm{d}\mathbf{u}_i f_i(\lambda_i)\mathbf{u}_i^\top,
\end{aligned}$$

where

$$\mathrm{d}\mathbf{u}_i f_i(\lambda)_i\mathbf{u}_i^\top = -\sum_{j=1,j\neq i}^{n} \mathbf{u}_j \frac{f_i(\lambda_i)}{\lambda_j - \lambda_i}\mathbf{u}_j^\top\mathrm{d}\mathbf{X}\mathbf{u}_i\mathbf{u}_i^\top$$

and

$$\mathbf{u}_i f_i(\lambda)_i\mathrm{d}\mathbf{u}_i^\top = -\sum_{j=1,j\neq i}^{n} \mathbf{u}_i \frac{f_i(\lambda)_i}{\lambda_j - \lambda_i}\mathbf{u}_i^\top\mathrm{d}\mathbf{X}\mathbf{u}_j\mathbf{u}_j^\top.$$

Thus, we have finally

$$\begin{aligned}
\mathrm{d}F(\mathbf{X}) &= \sum_{i=1}^{n} \mathbf{u}_i f_i'(\lambda_i)\mathbf{u}_i^\top\mathrm{d}\mathbf{X}\mathbf{u}_i\mathbf{u}_i^\top + \sum_{j=1,j\neq i}^{n} \frac{f_i(\lambda_i)}{\lambda_i - \lambda_j}\left(\mathbf{u}_j\mathbf{u}_j^\top\mathrm{d}\mathbf{X}\mathbf{u}_i\mathbf{u}_i^\top + \mathbf{u}_i\mathbf{u}_i^\top\mathrm{d}\mathbf{X}\mathbf{u}_j\mathbf{u}_j^\top\right) \\
&= \sum_{i=1}^{n}\left(\mathbf{u}_i f_i'(\lambda_i)\mathbf{u}_i^\top + \sum_{j=1,j\neq i}^{n} \mathbf{u}_j \frac{f_i(\lambda_i) - f_j(\lambda_j)}{\lambda_i - \lambda_j}\mathbf{u}_j^\top\right)\mathrm{d}\mathbf{X}\mathbf{u}_i\mathbf{u}_i^\top.
\end{aligned}$$

For $f_i(\lambda_i) = \lambda_i$ we find $\mathrm{d}\mathbf{X} = \mathrm{d}\mathbf{X}$. Using $f_i(\lambda_i) = \lambda_i^{-1}$, $f_i'(\lambda_i) = -\lambda_i^{-2}$ and hence $F(\mathbf{X}) = \mathbf{X}^{-1}$, we get

$$\begin{aligned}
\mathrm{d}F(\mathbf{X}) &= \sum_{i=1}^{n}\left(-\mathbf{u}_i\frac{1}{\lambda_i^2}\mathbf{u}_i^\top + \sum_{j=1,j\neq i}^{n} \mathbf{u}_j \frac{\lambda_i^{-1} - \lambda_j^{-1}}{\lambda_i - \lambda_j}\mathbf{u}_j^\top\right)\mathrm{d}\mathbf{X}\mathbf{u}_i\mathbf{u}_i^\top \\
&= -\sum_{i=1}^{n}\left(\mathbf{u}_i\frac{1}{\lambda_i}\mathbf{u}_i^\top + \sum_{j=1,j\neq i}^{n} \mathbf{u}_j \frac{1 - \lambda_j^{-1}\lambda_i}{\lambda_j - \lambda_i}\mathbf{u}_j^\top\right)\mathrm{d}\mathbf{X}\mathbf{u}_i\frac{1}{\lambda_i}\mathbf{u}_i^\top \\
&= -\sum_{i=1}^{n}\left(\mathbf{u}_i\frac{1}{\lambda_i}\mathbf{u}_i^\top + \sum_{j=1,j\neq i}^{n} \mathbf{u}_j\frac{1}{\lambda_j}\mathbf{u}_j^\top\right)\mathrm{d}\mathbf{X}\mathbf{u}_i\frac{1}{\lambda_i}\mathbf{u}_i^\top \\
&= -\sum_{i=j}^{n}\mathbf{u}_j\frac{1}{\lambda_j}\mathbf{u}_j^\top\mathrm{d}\mathbf{X}\sum_{i=1}^{n}\mathbf{u}_i\frac{1}{\lambda_i}\mathbf{u}_i^\top = -\mathbf{X}^{-1}\mathrm{d}\mathbf{X}\mathbf{X}^{-1}.
\end{aligned}$$

Appendix B

Convexity and Convex (Fenchel) duality

Convex sets, functions and their duality properties are very important concepts in analysis and optimisation [Boyd and Vandenberghe, 2004, Rockafellar, 1970] since they allow for strong statements about their behaviour. Convexity constrains the mathematical objects so that many local properties also hold globally.

B.1 Convex sets

A subset $\mathcal{X}$ of a vector space is called convex if every pair of objects $\mathbf{x}, \mathbf{z} \in \mathcal{X}$ can be connected by a line that is contained in $\mathcal{X}$. Formally, we have

$$\mathcal{X} \text{ convex if } \forall \mathbf{x}, \mathbf{z} \in \mathcal{X} \; \forall t \in [0,1] : \; (1-t)\mathbf{x} + t\mathbf{z} \in \mathcal{X}. \tag{B.1}$$

Convex sets are closed under intersection meaning that from convexity of $\mathcal{X}, \mathcal{Z}$ it follows that $\mathcal{X} \cap \mathcal{Z}$ is convex. Furthermore, convex combinations of elements are contained in convex sets. If $\mathbf{x}_i \in \mathcal{X}$, $i = 1..n$ and $\sum_{i=1}^n \alpha_i = 1$, $\alpha_i \geq 0$ then $\sum_{i=1}^n \alpha_i \mathbf{x}_i \in \mathcal{X}$.

B.2 Convex functions

The most appealing property of convex functions from an optimisation viewpoint is the fact that local minima correspond to global minima. Along these lines, the common wisdom in machine learning is that convex optimisation is easy and therefore considered a very desirable property. Convex functions are functions that can be lower-bounded by linear functions

$$f : \mathcal{X} \to \mathbb{R} \text{ convex if } \forall \mathbf{x}, \mathbf{z} \in \mathcal{X} \; \forall t \in [0,1] : \; f\left((1-t)\mathbf{x} + t\mathbf{z}\right) \leq (1-t) f(\mathbf{x}) + t f(\mathbf{z}). \tag{B.2}$$

A more general version of equation B.2 is known as Jensen's inequality

$$\begin{aligned} f\left(\sum_{i=1}^n p_i \mathbf{x}_i\right) &\leq \sum_{i=1}^n p_i f(\mathbf{x}_i), \; p_i \geq 0, \; \sum_{i=1}^n p_i = 1 \qquad \text{(B.3)} \\ f\left(\int \mathbb{P}(\mathbf{x})\mathbf{x} d\mathbf{x}\right) = f\left(\mathbb{E}_{\mathbb{P}(\mathbf{x})}[\mathbf{x}]\right) &\leq \int \mathbb{P}(\mathbf{x}) f(\mathbf{x})\, d\mathbf{x} = \mathbb{E}_{\mathbb{P}(\mathbf{x})}[f(\mathbf{x})], \; \mathbb{P}(\mathbf{x}) \geq 0, \; \int \mathbb{P}(\mathbf{x}) d\mathbf{x} \geq 0 \end{aligned}$$

that can be used to upper bound convex functions of linear combinations and expectations. Convexity of twice continuously differentiable functions is equivalent to a positive semidefinite Hessian matrix

$$f : \mathcal{X} \to \mathbb{R} \text{ convex if } \forall \mathbf{x} \in \mathcal{X} : \; \frac{\partial^2 f(\mathbf{x})}{\partial \mathbf{x} \partial \mathbf{x}^\top} \succcurlyeq \mathbf{0}.$$

Strict convexity requires $f\left((1-t)\mathbf{x} + t\mathbf{z}\right) < (1-t) f(\mathbf{x}) + t f(\mathbf{z})$. A function f is concave if $-f$ is convex.

The set of convex functions is closed under several operations [Boyd and Vandenberghe, 2004, 3.2] such as

- addition: $f(\mathbf{x}), g(\mathbf{x})$ convex $\Rightarrow f(\mathbf{x}) + g(\mathbf{x})$ convex
- positive scaling: $f(\mathbf{x})$ convex, $\alpha \in \mathbb{R}_+ \Rightarrow \alpha f(\mathbf{x})$ convex
- affine composition: $f(\mathbf{x})$ convex $\Rightarrow f(\mathbf{Az}+\mathbf{b})$ convex in $\mathbf{z}$
- pointwise maximisation: $f(\mathbf{x}), g(\mathbf{x})$ convex $\Rightarrow \max\{f(\mathbf{x}), g(\mathbf{x})\}$ convex and
- marginalisation: $f(\mathbf{x},\mathbf{z})$ jointly convex in $[\mathbf{x};\mathbf{z}] \Rightarrow \min_{\mathbf{x}} f(\mathbf{x},\mathbf{z})$ convex in $\mathbf{z}$.

B.3 Convex duality

Since convex functions can be lower bounded by linear functions, one can represent them as a maximum over linear functions with normal vector $\mathbf{z}$ and offset $f^\star(\mathbf{z})$

$$f(\mathbf{x}) = \max_{\mathbf{z}} \mathbf{z}^\top \mathbf{x} - f^\star(\mathbf{z}). \tag{B.4}$$

On an abstract level, $f(\mathbf{x})$ can be equivalently represented by points $(\mathbf{x}, f(\mathbf{x}))$ or by hyperplanes $\mathbf{z}^\top\mathbf{x} - f^\star(\mathbf{z})$. This is the duality relationship at the core of convex duality. The function $f^\star(\mathbf{z})$ is called the Legendre dual of $f(\mathbf{x})$. For strictly convex functions, we have $f^{\star\star} = f$. The duality relationship can be used to obtain lower bounds on the function $f(\mathbf{x})$

$$f(\mathbf{x}) \geq \mathbf{z}^\top\mathbf{x} - f^\star(\mathbf{z}) \overset{c}{=} \mathbf{z}^\top\mathbf{x}\,\forall\mathbf{z}. \tag{B.5}$$

For a point $\mathbf{x}$, the bound becomes tight, i.e. $f(\mathbf{x}) = \mathbf{z}^\top\mathbf{x} - f^\star(\mathbf{z})$ if $\mathbf{z} = \frac{\partial f(\mathbf{x})}{\partial \mathbf{x}}$. Similarly, concave functions can be upper bounded by linear functions, which turns out helpful in convex optimisation, where one can replace concave terms in the objective functions by simple linear functions as suggested by equation B.5.

B.4 Examples

In the following, we will provide some common duality relationships. In general, from the pair $f(\mathbf{x}) \mapsto f^\star(\mathbf{z})$, we can deduce the following variational representations of $f(\mathbf{x})$

$$\begin{array}{ll} f(\mathbf{x}) \text{ convex} & f(\mathbf{x}) = \max_{\mathbf{z}} \mathbf{z}^\top\mathbf{x} - f^\star(\mathbf{z}) \\ f(\mathbf{x}) \text{ concave} & f(\mathbf{x}) = \min_{\mathbf{z}} \mathbf{z}^\top\mathbf{x} - f^\star(\mathbf{z}). \end{array}$$

The following table lists useful pairs of functions and their respective Legendre duals.

function	$\mu f(\mathbf{x})$	$f(\mu\mathbf{x})$	$f(\mathbf{x})+a$	$f(\mathbf{x}+\mathbf{y})$	$\frac{1}{2}\mathbf{x}^\top\mathbf{Ax}$	e^x
Legendre dual	$\mu f^\star(\mathbf{z}/\mu)$	$f^\star(\mathbf{z}/\mu)$	$f^\star(\mathbf{z})-a$	$f^\star(\mathbf{z})-\mathbf{z}^\top\mathbf{y}$	$\frac{1}{2}\mathbf{z}^\top\mathbf{A}^{-1}\mathbf{z}$	$x\ln x - x$

For quadratic functions, we can obtain the following variational representations

$$\begin{aligned} \frac{1}{2}\mathbf{x}^\top\mathbf{A}^{-1}\mathbf{x} &= \max_{\mathbf{z}} \mathbf{x}^\top\mathbf{z} - \frac{1}{2}\mathbf{z}^\top\mathbf{Az} \text{ and} \\ -\mathbf{x}^\top\mathbf{A}^{-1}\mathbf{x} &= \min_{\mathbf{z}} \mathbf{z}^\top\mathbf{Az} - 2\mathbf{x}^\top\mathbf{z}. \end{aligned}$$

Appendix C

The Multivariate Gaussian

The multivariate Gaussian distribution is the analytically most convenient and therefore most important multivariate distribution for continuous variables. Besides being the maximum-entropy distribution for a fixed mean $\boldsymbol{\mu}$ and variance $\boldsymbol{\Sigma}$, the Gaussian family is closed under affine transformations, marginalisation and conditioning. Furthermore, the Gaussian distribution naturally emerges from the central limit theorem as the asymptotic distribution of sums of many random variables.

C.1 Gaussian density

The Gaussian distribution with mean $\boldsymbol{\mu} \in \mathbb{R}^n$, positive definite variance $\boldsymbol{\Sigma} \succ \mathbf{0} \in \mathbb{R}^{n\times n}$ has the density

$$\mathcal{N}(\mathbf{x}|\boldsymbol{\mu},\boldsymbol{\Sigma}) := \mathbb{P}(\mathbf{x}) = |2\pi\boldsymbol{\Sigma}|^{-\frac{1}{2}} \exp\left(-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu})^\top\boldsymbol{\Sigma}^{-1}(\mathbf{x}-\boldsymbol{\mu})\right).$$

Its marginals are given by $\mathbb{P}(x_i) = \mathcal{N}(x_i|\mu_i,\sigma_i^2)$, where $\boldsymbol{\sigma}^2 = \mathrm{dg}(\boldsymbol{\Sigma})$. Affine transformations of Gaussians produce Gaussians

$$\mathbf{x} \sim \mathcal{N}(\boldsymbol{\mu},\boldsymbol{\Sigma}) \Rightarrow \mathbf{B}\mathbf{x}+\mathbf{c} \sim \mathcal{N}(\mathbf{B}\boldsymbol{\mu}+\mathbf{c},\mathbf{B}\boldsymbol{\Sigma}\mathbf{B}^\top).$$

C.2 Unnormalised Gaussian

A second parametrisation of the distributions is given by the natural parameters $[\mathbf{b},\mathbf{A}]$, where $\mathbf{A}$ is the precision matrix. One can easily transform between the moment and the natural parameters via $[\boldsymbol{\mu},\boldsymbol{\Sigma}] = [\mathbf{A}^{-1}\mathbf{b},\mathbf{A}^{-1}]$. Often in calculations, the Gaussian components need not to be normalisable. Therefore, we work with Gaussian functions

$$\mathcal{G}(\mathbf{x}|\mathbf{b},\mathbf{A}) = \exp\left(\mathbf{b}^\top\mathbf{x}-\frac{1}{2}\mathbf{x}^\top\mathbf{A}\mathbf{x}\right),\ \mathbf{A}\succ\mathbf{0}.$$

Conditionals are best computed from the natural parametrisation

$$x_i|x_j \sim \frac{1}{Z}\mathcal{G}(x_i|b_j + A_{ij}x_j, A_{ii}).$$

C.3 Exponential family

A widely used class of distributions also including the Gaussian, is the *exponential family*

$$\mathbb{P}(\mathbf{x}|\boldsymbol{\theta}) = \exp\left(\boldsymbol{\theta}^\top\boldsymbol{\phi}(\mathbf{x}) - \Phi(\boldsymbol{\theta})\right),$$

where $\boldsymbol{\theta}$ denotes the natural or exponential parameters, $\boldsymbol{\phi}(\mathbf{x})$ is the vector of sufficient statistics and $\Phi(\boldsymbol{\theta}) = \ln\int\exp\left(\boldsymbol{\theta}^\top\boldsymbol{\phi}(\mathbf{x})\right)\mathrm{d}\mathbf{x}$ is the convex log partition function making sure that $\mathbb{P}(\mathbf{x}|\boldsymbol{\theta})$

integrates to 1. In the statistics literature [Wasserman, 2005, chapter 19], the equivalent term *log-linear models* is used. The vector $\boldsymbol{\eta} = \int \mathbb{P}(\mathbf{x}|\boldsymbol{\theta})\boldsymbol{\phi}(\mathbf{x})\mathrm{d}\mathbf{x}$ contains the moment parameters.

The Gaussian distribution can be obtained from the sufficient statistics $\boldsymbol{\phi}(\mathbf{x}) = [\mathbf{x}, \mathbf{x}\mathbf{x}^\top]$ and exponential parameters $\boldsymbol{\theta} = [\boldsymbol{\Sigma}^{-1}\boldsymbol{\mu}, -\frac{1}{2}\boldsymbol{\Sigma}^{-1}] = [\mathbf{b}, -\frac{1}{2}\mathbf{A}]$. The moment parameters are $\boldsymbol{\eta} = [\boldsymbol{\mu}, \boldsymbol{\Sigma} + \boldsymbol{\mu}\boldsymbol{\mu}^\top]$ and the log partition function $\Phi(\boldsymbol{\theta})$, jointly convex in $[\mathbf{b}, -\mathbf{A}]$ and $[\mathbf{b}, \mathbf{A}]$, reads $\Phi(\boldsymbol{\theta}) = \frac{1}{2}(\boldsymbol{\mu}^\top\boldsymbol{\Sigma}^{-1}\boldsymbol{\mu} + \ln|2\pi\boldsymbol{\Sigma}|) = \frac{1}{2}(\mathbf{b}^\top\mathbf{A}^{-1}\mathbf{b} - \ln|\mathbf{A}|) + \frac{n}{2}\ln 2\pi$.

C.4 Log partition function

Besides acting as a normaliser, the log partition function $\Phi(\boldsymbol{\theta})$ is closely related to the cumulant generating function; moments of $\mathbb{P}(\mathbf{x}|\boldsymbol{\theta})$ can be obtained by differentiation [Wainwright and Jordan, 2008, chapter 3]

$$\begin{aligned}
\frac{\partial}{\partial\boldsymbol{\theta}}\Phi(\boldsymbol{\theta}) &= \mathbb{E}_{\mathbb{P}(\mathbf{x}|\boldsymbol{\theta})}[\boldsymbol{\phi}(\mathbf{x})] = \int \mathbb{P}(\mathbf{x}|\boldsymbol{\theta})\boldsymbol{\phi}(\mathbf{x})\mathrm{d}\mathbf{x} = \boldsymbol{\eta} \\
\frac{\partial^2}{\partial\boldsymbol{\theta}\partial\boldsymbol{\theta}^\top}\Phi(\boldsymbol{\theta}) &= \mathbb{V}_{\mathbb{P}(\mathbf{x}|\boldsymbol{\theta})}[\boldsymbol{\phi}(\mathbf{x})] = \mathbb{E}_{\mathbb{P}(\mathbf{x}|\boldsymbol{\theta})}[(\boldsymbol{\phi}(\mathbf{x}) - \boldsymbol{\eta})(\boldsymbol{\phi}(\mathbf{x}) - \boldsymbol{\eta})^\top],
\end{aligned}$$

which nicely relates the moment $\boldsymbol{\eta}$ and the exponential parameters $\boldsymbol{\theta}$ via $\frac{\partial}{\partial\boldsymbol{\theta}}\Phi(\boldsymbol{\theta}) = \boldsymbol{\eta}$.

For the Gaussian distribution, we obtain

$$\begin{aligned}
\Phi(\mathbf{A}, \mathbf{b}) : &= \ln\int \mathcal{G}(\mathbf{x}|\mathbf{b}, \mathbf{A})\mathrm{d}\mathbf{x} = \ln\int \exp\left(-\frac{1}{2}(\mathbf{x}^\top\mathbf{A}\mathbf{x} - 2\mathbf{b}^\top\mathbf{A}^{-1}\mathbf{A}\mathbf{x})\right)\mathrm{d}\mathbf{x} \\
&= \frac{1}{2}\mathbf{b}^\top\mathbf{A}^{-1}\mathbf{b} + \ln\int \exp\left(-\frac{1}{2}(\mathbf{x} - \mathbf{A}^{-1}\mathbf{b})^\top\mathbf{A}(\mathbf{x} - \mathbf{A}^{-1}\mathbf{b})\right)\mathrm{d}\mathbf{x} \\
&= \frac{1}{2}\mathbf{b}^\top\mathbf{A}^{-1}\mathbf{b} + \frac{n}{2}\ln 2\pi - \frac{1}{2}\ln|\mathbf{A}| \text{ and}
\end{aligned}$$

$$\mathcal{N}(\mathbf{x}|\mathbf{A}^{-1}\mathbf{b}, \mathbf{A}^{-1}) = e^{-\Phi(\mathbf{A},\mathbf{b})}\mathcal{G}(\mathbf{x}|\mathbf{b}, \mathbf{A}).$$

Using convex duality, we can write

$$-2\Phi(\mathbf{A}, \mathbf{b}) = \ln|\mathbf{A}| + \min_{\mathbf{u}}\left[\mathbf{u}^\top\mathbf{A}\mathbf{u} - 2\mathbf{b}^\top\mathbf{u}\right] - n\ln 2\pi.$$

Also, $\exp\left(\frac{1}{2}\mathbf{b}^\top\mathbf{A}^{-1}\mathbf{b}\right) = \max_{\mathbf{u}}\exp\left(\mathbf{b}^\top\mathbf{u} - \frac{1}{2}\mathbf{u}^\top\mathbf{A}\mathbf{u}\right) = \max_{\mathbf{u}}\mathcal{G}(\mathbf{u}|\mathbf{b}, \mathbf{A})$ leads to

$$\int \mathcal{G}(\mathbf{x}|\mathbf{b}, \mathbf{A})\mathrm{d}\mathbf{x} = \sqrt{|2\pi\mathbf{A}^{-1}|}\exp\left(\frac{1}{2}\mathbf{b}^\top\mathbf{A}^{-1}\mathbf{b}\right) = \sqrt{|2\pi\mathbf{A}^{-1}|}\max_{\mathbf{x}}\mathcal{G}(\mathbf{x}|\mathbf{b}, \mathbf{A}).$$

Another useful identity for $\mathbf{b} = \mathbf{0}$ characterises the log determinant as a Gaussian integral

$$\ln|\mathbf{A}| = n\ln 2\pi - 2\Phi(\mathbf{A}, \mathbf{0}) = n\ln 2\pi - 2\ln\int \exp\left(-\frac{1}{2}\mathbf{u}^\top\mathbf{A}\mathbf{u}\right)\mathrm{d}\mathbf{u}. \tag{C.1}$$

C.5 Entropy

Finally, the entropy of a Gaussian variable is given by

$$\begin{aligned}
\mathcal{H}[\mathbb{P}(x)] &= \mathbb{E}_{\mathbb{P}(x)}[-\ln\mathbb{P}(x)] = -\int \mathbb{P}(x)\ln\mathbb{P}(x)\mathrm{d}x \\
\Rightarrow \mathcal{H}[\mathcal{N}(\mathbf{x}|\boldsymbol{\mu}, \boldsymbol{\Sigma})] &= \frac{1}{2}\ln|\boldsymbol{\Sigma}| + \frac{n}{2}(1 + \ln 2\pi).
\end{aligned}$$

C.6 Relative entropy

The Kullback-Leibler (KL) divergence to a Gaussian is obtained as

$$\begin{aligned}\mathrm{KL}\left(\mathbb{Q}(\mathbf{x})||\mathbb{P}(\mathbf{x})\right) &= \int \mathbb{Q}(\mathbf{x})\ln\frac{\mathbb{Q}(\mathbf{x})}{\mathbb{P}(\mathbf{x})}\mathrm{d}\mathbf{x} = -\mathcal{H}\left[\mathbb{P}(x)\right] - \mathbb{E}_{\mathbb{Q}(x)}\left[\ln\mathbb{P}(x)\right] \\ \Rightarrow \mathrm{KL}\left(\mathcal{N}(\mathbf{x}|\boldsymbol{\mu},\boldsymbol{\Sigma})||\mathbb{P}(\mathbf{x})\right) &= -\frac{1}{2}\ln|\boldsymbol{\Sigma}| - \frac{n}{2}(1+\ln 2\pi) - \int \mathcal{N}(\mathbf{x}|\boldsymbol{\mu},\boldsymbol{\Sigma})\ln\mathbb{P}(\mathbf{x})\mathrm{d}\mathbf{x}.\end{aligned}$$

The relative entropy $\mathrm{KL}\left(\mathbb{P}(\mathbf{x}|\boldsymbol{\theta})||\mathbb{P}(\mathbf{x}|\tilde{\boldsymbol{\theta}})\right) = \mathrm{KL}\left(\boldsymbol{\theta}||\tilde{\boldsymbol{\theta}}\right)$ can be expressed using the moment parameters $\boldsymbol{\eta}$ and the exponential parameters $\boldsymbol{\theta}$

$$\begin{aligned}\mathrm{KL}\left(\boldsymbol{\theta}||\tilde{\boldsymbol{\theta}}\right) &= \boldsymbol{\eta}^\top(\boldsymbol{\theta}-\tilde{\boldsymbol{\theta}}) + \Phi(\tilde{\boldsymbol{\theta}}) - \Phi(\boldsymbol{\theta}) \\ &= \tilde{\boldsymbol{\theta}}^\top(\tilde{\boldsymbol{\eta}}-\boldsymbol{\eta}) + \Phi^\star(\boldsymbol{\eta}) - \Phi^\star(\tilde{\boldsymbol{\eta}}) \;=\; \mathrm{KL}\left(\boldsymbol{\eta}||\tilde{\boldsymbol{\eta}}\right),\end{aligned}$$

where $\Phi^\star(\boldsymbol{\eta})$ is the convex conjugate of $\Phi(\boldsymbol{\theta})$[1]. As a result, we see that $\mathrm{KL}\left(\boldsymbol{\theta}||\tilde{\boldsymbol{\theta}}\right) = \mathrm{KL}\left(\boldsymbol{\eta}||\tilde{\boldsymbol{\eta}}\right)$ is both convex in $\tilde{\boldsymbol{\theta}}$ and $\boldsymbol{\eta}$ [Seeger, 2003, A.13].

Thus, the Gaussian relative entropy $\mathrm{KL}\left(\mathcal{N}(\mathbf{x}|\boldsymbol{\mu},\boldsymbol{\Sigma})||\mathbb{P}(\mathbf{x})\right)$ is jointly convex in $[\boldsymbol{\mu},\boldsymbol{\Sigma}+\boldsymbol{\mu}\boldsymbol{\mu}^\top]$ or convex in $\boldsymbol{\Sigma}$ for $\boldsymbol{\mu}=\mathbf{0}$.

Furthermore, the relative entropy $\mathrm{KL}\left(\mathcal{N}_1||\mathcal{N}_2\right)$ between two Gaussians

$$\begin{aligned}2\cdot\mathrm{KL}\left(\mathcal{N}_1||\mathcal{N}_2\right) &= -\ln\left|\boldsymbol{\Sigma}_1\boldsymbol{\Sigma}_2^{-1}\right| + \mathrm{tr}\left(\boldsymbol{\Sigma}_1\boldsymbol{\Sigma}_2^{-1}-\mathbf{I}\right) + (\boldsymbol{\mu}_1-\boldsymbol{\mu}_2)^\top\boldsymbol{\Sigma}_2^{-1}(\boldsymbol{\mu}_1-\boldsymbol{\mu}_2) \\ &= -\ln\left|\mathbf{A}_1^{-1}\mathbf{A}_2\right| + \mathrm{tr}\left((\boldsymbol{\Sigma}_1+\boldsymbol{\mu}_1\boldsymbol{\mu}_1^\top)\mathbf{A}_2-\mathbf{I}\right) - 2\mathbf{b}_2^\top\boldsymbol{\mu}_1 + \mathbf{b}_2^\top\mathbf{A}_2^{-1}\mathbf{b}_2\end{aligned}$$

is (interestingly) jointly convex in $[\boldsymbol{\mu}_1,\boldsymbol{\Sigma}_1]$ and $[\mathbf{b}_2,\mathbf{A}_2]$.

C.7 Gaussian measure of convex functions

The integral of the negative log potential $f(s) = -\ln\mathcal{T}(s)$ w.r.t. a general Gaussian $\mathcal{N}(s|\mu,\sigma^2)$

$$\omega(\mu,\sigma^2) = \int\mathcal{N}(s|\mu,\sigma^2)f(s)\,\mathrm{d}s = \int\mathcal{N}(s)f(\sigma s+\mu)\,\mathrm{d}s$$

occurs in the KL objective to be minimised in equation 2.15. In the following, we exploit the Leibniz integral rule

$$\frac{\mathrm{d}}{\mathrm{d}\alpha}\int_a^b f(x,\alpha)\mathrm{d}x = \int_a^b\frac{\partial}{\partial\alpha}f(x,\alpha)\mathrm{d}x + f(b,\alpha)\frac{\partial b}{\partial\alpha} - f(a,\alpha)\frac{\partial a}{\partial\alpha}$$

to show that $\omega(\mu,\sigma^2)$ is convex in μ and σ whenever $f(s)$ is convex itself. Further, we provide an example showing that $\omega(\mu,\sigma^2)$ is not convex in $\nu=\sigma^2$ in general.

We start by showing that $\omega(\mu,\sigma^2)$ is convex in the mean μ

$$\begin{aligned}\omega_\mu = \frac{\partial\omega}{\partial\mu} &= \int\mathcal{N}(s)f'(\sigma s+\mu)\,\mathrm{d}s \geq 0 \\ \omega_{\mu\mu} = \frac{\partial^2\omega}{\partial\mu^2} &= \int\mathcal{N}(s)f''(\sigma s+\mu)\,\mathrm{d}s \geq 0 \Leftarrow f''(s)\geq 0\end{aligned}$$

and in the standard deviation σ

$$\begin{aligned}\omega_\sigma = \frac{\partial\omega}{\partial\sigma} &= \int\mathcal{N}(s)sf'(\sigma s+\mu)\,\mathrm{d}s \\ \omega_{\sigma\sigma} = \frac{\partial^2\omega}{\partial\sigma^2} &= \int\mathcal{N}(s)s^2f''(\sigma s+\mu)\,\mathrm{d}s \geq 0 \Leftarrow f''(s)\geq 0.\end{aligned}$$

[1] $\Phi^\star(\boldsymbol{\eta})$ is roughly equal to the negative entropy of $\mathbb{P}(\mathbf{x}|\boldsymbol{\eta})$ Wainwright and Jordan [2008, chapter 3.6].

One can even show joint convexity in (μ,σ) by computing

$$\omega_{\mu\sigma}=\omega_{\sigma\mu}=\int\mathcal{N}(s)sf''\left(\sigma s+\mu\right)\mathrm{d}s$$

and showing that the determinant $D=|\mathbf{H}_\omega|$ of the Hessian $\mathbf{H}_\omega=\begin{bmatrix}\omega_{\mu\mu} & \omega_{\sigma\mu}\\ \omega_{\sigma\mu} & \omega_{\sigma\sigma}\end{bmatrix}$ is non-negative. We notice that all components of the Hessian $\mathbf{H}_\omega$ have the form $\eta_k=\int s^k\pi(s)\mathrm{d}s$ for $k=0,1,2$ and $\pi(s)=\mathcal{N}(s)f''\left(\sigma s+\mu\right)\geq 0\Leftarrow f''(s)\geq 0$. We write

$$D=\omega_{\sigma\sigma}\omega_{\mu\mu}-\omega_{\sigma\mu}^2=\eta_2\eta_0-\eta_1^2$$

and notice immediately $D\geq 0$ because the variance v of a random variable with density $\frac{\pi(s)}{\eta_0}$ can be expressed as

$$v=\frac{\eta_2}{\eta_0}-\frac{\eta_1^2}{\eta_0^2}=\frac{D}{\eta_0^2}\geq 0.$$

From now on, we restrict ourselves to Laplace potentials $f(s)=-\ln\mathcal{T}(s)=|s|$. From $\frac{\partial}{\partial s}\mathcal{N}(s)=-s\mathcal{N}(s)$, we find $\int_{s_0}^\infty s\mathcal{N}(s)\mathrm{d}s=\mathcal{N}(s_0)$ and using $\frac{\partial}{\partial s}\mathcal{N}(s|\mu,\sigma^2)=\frac{\mu-s}{\sigma}\mathcal{N}(s|\mu,\sigma^2)$, we can deduce $\int_0^\infty s\mathcal{N}(s|\mu,\sigma^2)\mathrm{d}s=\mu\Phi(\mu/\sigma)+\sigma\mathcal{N}(\mu/\sigma)$, which yields

$$\begin{aligned}\omega_L(\mu,\sigma^2)=\int_{-\infty}^\infty\mathcal{N}(s|\mu,\sigma^2)|s|\mathrm{d}s &= \int_0^\infty\left[\mathcal{N}(s|\mu,\sigma^2)+\mathcal{N}(s|-\mu,\sigma^2)\right]s\mathrm{d}s\\ &= \mu\Phi\left(\frac{\mu}{\sigma}\right)+\sigma\mathcal{N}\left(\frac{\mu}{\sigma}\right)-\mu\Phi\left(-\frac{\mu}{\sigma}\right)+\sigma\mathcal{N}\left(-\frac{\mu}{\sigma}\right)\\ &= 2\mu\left[\Phi\left(\frac{\mu}{\sigma}\right)-\frac{1}{2}\right]+2\sigma\mathcal{N}\left(\frac{\mu}{\sigma}\right).\end{aligned}$$

Already $\omega_L(0,\nu)=\sqrt{\frac{2}{\pi}\nu}$ is not convex in $\nu=\sigma^2$.

C.8 Non-convex relative entropy

We pick a 1d log-concave exponential family model $\mathbb{P}(u)$ with Laplace prior $\mathcal{T}(u)=\frac{1}{2}\exp(-|u|)$ and Gaussian likelihood $\mathcal{N}(u)$, i.e. $\mathbf{X}=\mathbf{B}=\sigma^2=1$ and $\mathbf{y}=0$, hence $\mathbb{P}(u)=\frac{1}{Z}\mathcal{N}(u)\mathcal{T}(u)$, $Z=\int\mathcal{N}(u)\mathcal{T}(u)\mathrm{d}u$. Further, we choose the class of Gaussians $\mathbb{Q}(u)=\mathcal{N}(u|\mu,\sigma^2)$ as approximating distribution.

With the equality from appendix C.7 in mind, the Kullback-Leibler divergence is given by

$$\begin{aligned}\mathrm{KL}(\mu,\sigma^2) &= \mathrm{KL}\left(\mathcal{N}(u|\mu,\sigma^2)||\frac{1}{Z}\mathcal{N}(u)\mathcal{T}(u)\right)=\int_{-\infty}^\infty\mathcal{N}(u|\mu,\sigma^2)\ln\frac{Z\mathcal{N}(u|\mu,\sigma^2)}{\mathcal{N}(u)\mathcal{T}(u)}\mathrm{d}u\\ &= -\mathcal{H}[\mathcal{N}(u|\mu,\sigma^2)]+\ln Z-\int_{-\infty}^\infty\mathcal{N}(u|\mu,\sigma^2)\ln\mathcal{N}(u)\mathrm{d}u-\int_{-\infty}^\infty\mathcal{N}(u|\mu,\sigma^2)\ln\mathcal{T}(u)\mathrm{d}u\\ &= C-\ln\sigma+\frac{\mu^2+\sigma^2}{2}+\int_{-\infty}^\infty\mathcal{N}(u|\mu,\sigma^2)|u|\mathrm{d}u,\ C=\ln Z-\frac{1}{2}+\ln 2\\ &= C-\ln\sigma+\frac{\mu^2+\sigma^2}{2}+\omega_L(\mu,\sigma^2).\end{aligned}$$

The general convexity result of $\omega(\mu,\sigma^2)$ from appendix C.7 implies that $\mathrm{KL}(\mu,\sigma^2)$ is jointly convex in (μ,σ).

However, already, the special case $\mathrm{KL}(0,\sigma^2)=C-\frac{1}{2}\ln\nu+\frac{\nu}{2}+\sqrt{\frac{2}{\pi}\nu}$ is not convex in $\nu=\sigma^2$ since the second derivative

$$\frac{\partial^2}{\partial\nu^2}\mathrm{KL}(0,\nu)=\frac{1}{2\nu^2}\left(1-\sqrt{\frac{\nu}{2\pi}}\right)$$

changes sign at $\nu = 2\pi$. Note that this is not in contradiction to the convexity statement in appendix C.6 since the distributions on the right and left side of the KL-divergence have different sufficient statistics, which clamps some natural parameters θ_i and $\tilde{\theta}_j$ to 0. As a result, the path in $\boldsymbol{\eta}$ becomes nonlinear; a contradiction requires non-convexity along a linear path.

Appendix D

Inference and Design in Linear Models

D.1 Reparametrisation rules

The following rules can be used to perform changes of variables.

$$\begin{aligned}\int_{\mathcal{U}} \phi(\mathbf{u})\mathrm{d}\mathbf{u} &= \int_{\xi^{-1}(\mathcal{U})} \phi\left(\boldsymbol{\xi}(\boldsymbol{\rho})\right) \left|\det\left(\frac{\partial \boldsymbol{\xi}(\boldsymbol{\rho})}{\partial \boldsymbol{\rho}^\top}\right)\right| \mathrm{d}\boldsymbol{\rho} \\ \mathbb{P}(\mathbf{u}) &= \mathbb{P}\left(\boldsymbol{\xi}(\boldsymbol{\rho})\right) \left|\det\left(\frac{\partial \boldsymbol{\xi}(\boldsymbol{\rho})}{\partial \boldsymbol{\rho}^\top}\right)\right| \\ \mathrm{d}\mathbf{u} &= \left|\det\left(\frac{\partial \boldsymbol{\xi}(\boldsymbol{\rho})}{\partial \boldsymbol{\rho}^\top}\right)\right| \mathrm{d}\boldsymbol{\rho} \\ \mathbb{P}(\mathbf{u})\mathrm{d}\mathbf{u} &= \mathbb{P}\left(\boldsymbol{\xi}(\boldsymbol{\rho})\right) \mathrm{d}\boldsymbol{\rho}\end{aligned}$$

D.2 Invariance of maximum likelihood estimation

We start from the original likelihood $\mathbb{P}(\mathbf{y}|\mathbf{u})$ and a likelihood $\mathbb{P}(\mathbf{y}|\boldsymbol{\xi}(\boldsymbol{\rho}))$ using a different coordinate system $\mathbf{u} = \boldsymbol{\xi}(\boldsymbol{\rho})$. The maximum likelihood estimators of $\mathbf{u}$ and $\boldsymbol{\rho}$ are related by

$$\hat{\mathbf{u}} = \arg\max_{\mathbf{u}} \mathbb{P}(\mathbf{y}|\mathbf{u}) = \boldsymbol{\xi}\left(\arg\max_{\boldsymbol{\rho}} \mathbb{P}(\mathbf{y}|\boldsymbol{\xi}(\boldsymbol{\rho}))\right) = \boldsymbol{\xi}\left(\hat{\boldsymbol{\rho}}\right) = \widehat{\boldsymbol{\xi}(\boldsymbol{\rho})}$$

implying that it does not matter whether we estimate the variance or the standard deviation of a random variable via maximum likelihood since they can be converted into one another post hoc.

The second type of invariance is about the data y_i, $i = 1..m$. By the transformation

$$\begin{aligned}\hat{\mathbf{u}} &= \arg\max_{\mathbf{u}} \mathbb{P}(\mathbf{y}|\mathbf{u}) = \arg\max_{\mathbf{u}} \mathbb{P}\left(\boldsymbol{\xi}(\boldsymbol{\rho})|\mathbf{u}\right) \left|\det\left(\frac{\partial \boldsymbol{\xi}(\boldsymbol{\rho})}{\partial \boldsymbol{\rho}^\top}\right)\right| \\ &= \arg\max_{\mathbf{u}} \mathbb{P}\left(\boldsymbol{\xi}(\boldsymbol{\rho})|\mathbf{u}\right) = \arg\max_{\mathbf{u}} \mathbb{P}_{\xi}\left(\boldsymbol{\rho}|\mathbf{u}\right),\end{aligned}$$

we can see that the estimate $\hat{\mathbf{u}}$ will be the same if we use a log-normal distribution $\mathbb{P}_\xi$ or a normal distribution $\mathbb{P}$ on the log of the data.

D.3 Invariance of Bayesian inference

Marginal likelihood

In addition to the invariances of general maximum likelihood estimation (appendix D.2), the marginal likelihood of a hyperparameter $\boldsymbol{\theta}$ is invariant to reparametrisation of the latent variables $\boldsymbol{\xi} : \boldsymbol{\rho} \mapsto \mathbf{u}$

$$\begin{aligned}
\mathbb{P}(\mathbf{y}|\boldsymbol{\theta}) &= \int \mathbb{P}(\mathbf{u})\mathbb{P}(\mathbf{y}|\mathbf{u})\mathrm{d}\mathbf{u} \\
&= \int \overbrace{\mathbb{P}\left(\boldsymbol{\xi}(\boldsymbol{\rho})\right)\left|\det\left(\frac{\partial\boldsymbol{\xi}(\boldsymbol{\rho})}{\partial\boldsymbol{\rho}^\top}\right)\right|}^{\mathbb{Q}(\boldsymbol{\rho})} \overbrace{\mathbb{P}\left(\mathbf{y}|\boldsymbol{\xi}(\boldsymbol{\rho})\right)}^{\mathbb{Q}(\mathbf{y}|\boldsymbol{\rho})} \overbrace{\left|\det\left(\frac{\partial\boldsymbol{\xi}(\boldsymbol{\rho})}{\partial\boldsymbol{\rho}^\top}\right)\right|^{-1}\mathrm{d}\mathbf{u}}^{\mathrm{d}\boldsymbol{\rho}} \\
&= \int \mathbb{Q}(\boldsymbol{\rho})\mathbb{Q}(\mathbf{y}|\boldsymbol{\rho})\mathrm{d}\boldsymbol{\rho} =: \mathbb{Q}(\mathbf{y}|\boldsymbol{\theta}).
\end{aligned}$$

Decision after inference

The posterior parametrised with $\mathbf{u} = \boldsymbol{\xi}(\boldsymbol{\rho})$ is given by

$$\begin{aligned}
\mathbb{P}(\mathbf{u}|\mathbf{y}) &= \frac{\mathbb{P}(\mathbf{u})\mathbb{P}(\mathbf{y}|\mathbf{u})}{\mathbb{P}(\mathbf{y})} = \frac{\mathbb{Q}(\boldsymbol{\rho})\mathbb{Q}(\mathbf{y}|\boldsymbol{\rho})}{\mathbb{Q}(\mathbf{y})}\left|\det\left(\frac{\partial\boldsymbol{\xi}(\boldsymbol{\rho})}{\partial\boldsymbol{\rho}^\top}\right)\right|^{-1} \\
&= \mathbb{Q}(\boldsymbol{\rho}|\mathbf{y})\left|\det\left(\frac{\partial\boldsymbol{\xi}(\boldsymbol{\rho})}{\partial\boldsymbol{\rho}^\top}\right)\right|^{-1},
\end{aligned}$$

therefore the Bayes estimator based on the minimum average loss

$$\begin{aligned}
\mathbf{u}^\star &= \arg\min_{\tilde{\mathbf{u}}} \int \ell(\mathbf{u},\tilde{\mathbf{u}})\mathbb{P}(\mathbf{u}|\mathbf{y})\mathrm{d}\mathbf{u} = \arg\min_{\tilde{\mathbf{u}}} \int \ell\left(\boldsymbol{\xi}(\boldsymbol{\rho}),\tilde{\mathbf{u}}\right)\mathbb{Q}(\boldsymbol{\rho}|\mathbf{y})\mathrm{d}\boldsymbol{\rho} \\
&= \arg\min_{\boldsymbol{\xi}(\tilde{\boldsymbol{\rho}})} \int \ell\left(\boldsymbol{\xi}(\boldsymbol{\rho}),\boldsymbol{\xi}(\tilde{\boldsymbol{\rho}})\right)\mathbb{Q}(\boldsymbol{\rho}|\mathbf{y})\mathrm{d}\boldsymbol{\rho} = \boldsymbol{\xi}\left(\arg\min_{\tilde{\boldsymbol{\rho}}} \int \ell_{\boldsymbol{\xi}}\left(\boldsymbol{\rho},\tilde{\boldsymbol{\rho}}\right)\mathbb{Q}(\boldsymbol{\rho}|\mathbf{y})\mathrm{d}\boldsymbol{\rho}\right) = \boldsymbol{\xi}(\boldsymbol{\rho}^\star)
\end{aligned}$$

as measured by the loss function $\ell(\mathbf{u},\tilde{\mathbf{u}})$ is invariant to a reparametrisation $\boldsymbol{\xi}$ if the loss is also transformed (into $\ell_{\boldsymbol{\xi}}$).

D.4 Cumulant based entropy approximation

Suppose, we are given an n-dimensional density $\mathbb{P}(\mathbf{x})$ with mean vector $\mathbf{m} = \mathbb{E}_{\mathbb{P}}[\mathbf{x}]$ and covariance matrix $\mathbf{V} = \mathbb{V}_{\mathbb{P}}[\mathbf{x}]$. A *moment* κ^{ijk} of $\mathbb{P}(\mathbf{x})$ is defined as the scalar expectation $\mathbb{E}_{\mathbb{P}}[x_i x_j x_k]$. A *cumulant* $\kappa^{i,j,k}$ [McCullagh, 1987, ch. 2] can be written in terms of the moments

$$\begin{aligned}
\kappa^{i,j} &= \kappa^{ij} - \kappa^i\kappa^j \\
\kappa^{i,j,k} &= \kappa^{ijk} - \kappa^{ij}\kappa^k - \kappa^{ik}\kappa^j - \kappa^i\kappa^{jk} + 2\kappa^i\kappa^j\kappa^k.
\end{aligned}$$

The Gram-Charlier A series [Barndorff-Nielsen and Cox, 1989] allows to expand the distribution $\mathbb{P}(\mathbf{x})$ in terms of the Gaussian $\mathcal{N}(\mathbf{x}|\mathbf{m},\mathbf{V})$ having the same mean and variance as $\mathbb{P}(\mathbf{x})$ and an infinite sum composed of sums cumulants of rising order weighted with Hermite polynomials $h_{ijk..}(\mathbf{x})$

$$\mathbb{P}(\mathbf{x}) = \mathcal{N}(\mathbf{x}|\mathbf{m},\mathbf{V})\left(1 + \frac{1}{3!}\sum_{i,j,k}\kappa^{i,j,k}h_{ijk}(\mathbf{x}) + \frac{1}{4!}\ldots\right).$$

This allows to approximate the differential entropy [Hulle, 2005] neglecting higher order terms by

$$\mathcal{H}[\mathbb{P}(\mathbf{x})] \approx \mathcal{H}[\mathcal{N}(\mathbf{x}|\mathbf{m},\mathbf{V})] - \frac{1}{12}\left(\sum_i(\tilde{\kappa}^{i,i,i})^2 + 3\sum_{i\neq j}(\tilde{\kappa}^{i,i,j})^2 + \frac{1}{6}\sum_{i<j<k}(\tilde{\kappa}^{i,j,k})^2\right),$$

where $\mathcal{H}[\mathcal{N}(\mathbf{x}|\mathbf{m},\mathbf{V})] = (\ln|\mathbf{V}| + n\ln 2\pi + n)/2$ is the entropy of the best Gaussian approximation, from which a sum of squared standardised cumulants $\tilde{\kappa}^{i,j,k} := \frac{\kappa^{i,j,k}}{\sigma_i\sigma_j\sigma_k} = \frac{\kappa^{i,j,k}}{\sqrt{\kappa^{i,i}\kappa^{j,j}\kappa^{k,k}}}$ is subtracted. This is effectively a decomposition of $\mathcal{H}[\mathbb{P}(\mathbf{x})]$ into a component based on the *scale* in terms of the Gaussian entropy and the *shape* given by the sum of third order standardised cumulants.

Appendix E

Convex Inference Relaxations and Algorithms

E.1 Convexity of log determinant

It is well known [Boyd and Vandenberghe, 2004], that $\boldsymbol{\gamma}^{-1} \mapsto \ln|\mathbf{A}|$ with $\mathbf{A} = \mathbf{X}^\top\mathbf{X} + \mathbf{B}^\top f(\mathbf{\Gamma})\mathbf{B}$ is concave for $f_j(\gamma_j) = \gamma_j^{-1}$. We will show that $\boldsymbol{\gamma} \mapsto \ln|\mathbf{A}|$ is convex whenever all scalar functions $\ln f_j(\gamma_j)$ are convex. We write $f(\boldsymbol{\gamma})$ or $f(\mathbf{\Gamma})$ to refer to the matrix with components $f_j(\gamma_j)$ on the diagonal. There is an elaborate way of getting the general result and a simple and intuitive way for a special case pointed out by Manfred Opper. We will first present the simple approach and then look at the general case.

First of all, log-convexity of $f(\gamma)$ is equivalent to $\frac{\mathrm{d}^2}{\mathrm{d}\gamma^2}\ln f(\gamma) \geq 0$ since $f(\gamma) \geq 0$ will be twice differentiable in the following.

$$\frac{\mathrm{d}\ln f(\gamma)}{\mathrm{d}\gamma} = \frac{f'(\gamma)}{f(\gamma)}, \; \frac{\mathrm{d}^2\ln f(\gamma)}{\mathrm{d}\gamma^2} = \frac{f''(\gamma)f(\gamma) - f'(\gamma)f'(\gamma)}{[f(\gamma)]^2} \geq 0 \Leftrightarrow f''(\gamma)f(\gamma) \geq [f'(\gamma)]^2 \quad \text{(E.1)}$$

Intuitive and simple approach

Making use of equation C.1, we can rewrite the log determinant as a negative Gaussian integral $\ln|\mathbf{A}| = n\ln 2\pi - 2\ln\int\exp(-\frac{1}{2}\mathbf{u}^\top\mathbf{A}\mathbf{u})\mathrm{d}\mathbf{u}$. If the map $(\mathbf{u},\boldsymbol{\gamma}) \mapsto \mathbf{u}^\top\mathbf{A}\mathbf{u}$ is jointly convex, then $(\mathbf{u},\boldsymbol{\gamma}) \mapsto \exp(-\frac{1}{2}\mathbf{u}^\top\mathbf{A}\mathbf{u})$ is jointly log-concave. The marginalisation theorem due to Prékopa [Bogachev, 1998, §1.8] states that marginals of log-concave functions are log-concave. Consequently, the Gaussian integral $\boldsymbol{\gamma} \mapsto \int\exp(-\frac{1}{2}\mathbf{u}^\top\mathbf{A}\mathbf{u})\mathrm{d}\mathbf{u}$ is log-concave implying convexity of $\boldsymbol{\gamma} \mapsto \ln|\mathbf{A}|$. So, when is $(\mathbf{u},\boldsymbol{\gamma}) \mapsto \mathbf{u}^\top\mathbf{A}\mathbf{u} = \mathbf{u}^\top\mathbf{X}^\top\mathbf{X}\mathbf{u} + \sum_{j=1}^q s_j^2 f(\gamma_j)$, $\mathbf{s} = \mathbf{B}\mathbf{u}$ jointly convex? Obviously, exactly if $(s,\gamma) \mapsto s^2 f(\gamma)$ is jointly convex since convex functions are closed under affine transformations of the input. Computing the determinant of the Hessian, we get $|\mathbf{H}_{s,\gamma}| = 2f(\gamma)\cdot s^2 f''(\gamma) - [2sf'(\gamma)]^2$, which is positive for $f(\gamma)\cdot f''(\gamma) \geq 2[f'(\gamma)]^2$. This condition is stricter than the one imposed by equation E.1, hence $f(\gamma) = \gamma^{-1}$ is covered but $f(\gamma) = e^\gamma$ not.

General case

We exploit the result that a a function $\phi(\boldsymbol{\gamma})$ is jointly convex in $\boldsymbol{\gamma}$ iff. ϕ is convex along all lines. Formally the scalar function $\phi(t) := \phi(\boldsymbol{\gamma}_t)$, $\boldsymbol{\gamma}_t := \mathbf{p} + t\cdot\mathbf{d} \geq 0$ has to be convex for all points $\mathbf{p} \in \mathbb{R}^q$ and directions $\mathbf{d} \in \mathbb{R}^q$. In order to show this, we will show that the second derivative

$\phi''(t)$ is always non-negative. We use the calculus of appendix A.1.

$$\begin{aligned}
\phi(t) &= \ln|\mathbf{A}_t| = \ln\left|\mathbf{X}^\top\mathbf{X} + \mathbf{B}^\top\mathbf{F}_t\mathbf{B}\right|,\ \mathbf{F}_t = f(\mathbf{p}+t\cdot\mathbf{d}) \in \mathbb{R}^{q\times q} \\
\mathrm{d}\phi(t) &= \operatorname{tr}\left(\mathbf{A}_t^{-1}\mathbf{B}^\top\mathbf{F}_t'\mathbf{D}\mathbf{B}\right)\mathrm{d}t,\ \mathbf{D} = \operatorname{diag}(\mathbf{d}),\ \mathbf{F}_t' = f'(\mathbf{p}+t\cdot\mathbf{d}) \\
\mathrm{d}^2\phi(t) &= \operatorname{tr}\left(\mathbf{D}\mathbf{B}\mathrm{d}\mathbf{A}_t^{-1}\mathbf{B}^\top\mathbf{F}_t' + \mathbf{D}\mathbf{B}\mathbf{A}_t^{-1}\mathbf{B}^\top\mathrm{d}\mathbf{F}_t'\right)\mathrm{d}t \\
&= \operatorname{tr}\left(\mathbf{D}\mathbf{B}\mathbf{A}_t^{-1}\left[-\mathrm{d}\mathbf{A}_t\mathbf{A}_t^{-1}\mathbf{B}^\top\mathbf{F}_t' + \mathbf{B}^\top\mathrm{d}\mathbf{F}_t'\right]\right)\mathrm{d}t \\
&= \operatorname{tr}\left(\mathbf{D}\mathbf{B}\mathbf{A}_t^{-1}\mathbf{B}^\top\left[-\mathbf{F}_t'\mathbf{D}\mathbf{B}\mathbf{A}_t^{-1}\mathbf{B}^\top\mathbf{F}_t' + \mathbf{F}_t''\mathbf{D}\right]\right)\mathrm{d}t^2,\ \mathbf{F}_t'' = f''(\mathbf{p}+t\cdot\mathbf{d}) \\
\phi''(t) &= \operatorname{tr}\left(\mathbf{D}\mathbf{S}_t\mathbf{D}\left[\mathbf{F}_t'' - \mathbf{F}_t'\mathbf{S}_t\mathbf{F}_t'\right]\right),\ \mathbf{S}_t = \mathbf{B}\mathbf{A}_t^{-1}\mathbf{B}^\top \\
&= \operatorname{tr}\left(\mathbf{F}_t'\mathbf{D}\mathbf{S}_t\mathbf{D}\mathbf{F}_t'\left[\mathbf{G}_t - \mathbf{S}_t\right]\right),\ \mathbf{G}_{t,jj} = \frac{f_j''(p_j+t\cdot d_j)}{\left[f_j'(p_j+t\cdot d_j)\right]^2}
\end{aligned}$$

Since $\mathbf{S}_t$ is symmetric positive semidefinite (spsd), $\mathbf{F}_t'\mathbf{D}\mathbf{S}_t\mathbf{D}\mathbf{F}_t'$ is also spsd and hence $\phi''(t)$ will be non-negative if the matrix $\mathbf{G}_t - \mathbf{S}_t$ is spsd, which is the case if $\mathbf{z}^\top(\mathbf{G}_t - \mathbf{S}_t)\mathbf{z} \geq 0$ for all $\mathbf{z}$.

$$\begin{aligned}
\mathbf{z}^\top(\mathbf{G}_t - \mathbf{S}_t)\mathbf{z} &= \mathbf{z}^\top\mathbf{G}_t\mathbf{z} - \mathbf{z}^\top\mathbf{B}\mathbf{A}_t^{-1}\mathbf{B}^\top\mathbf{z} \\
&= \mathbf{z}^\top\mathbf{G}_t\mathbf{z} + \left(\min_{\mathbf{u}}\mathbf{u}^\top\mathbf{A}_t\mathbf{u} - 2\mathbf{z}^\top\mathbf{B}\mathbf{u}\right) \\
&= \mathbf{z}^\top(\mathbf{F}_t')^{-1}\mathbf{F}_t''(\mathbf{F}_t')^{-1}\mathbf{z} + \left(\min_{\mathbf{u}}\mathbf{u}^\top\mathbf{X}^\top\mathbf{X}\mathbf{u} + \mathbf{u}\mathbf{B}^\top\mathbf{F}_t\mathbf{B}\mathbf{u} - 2\mathbf{z}^\top\mathbf{B}\mathbf{u}\right) \\
&\geq \mathbf{z}^\top(\mathbf{F}_t')^{-1}\mathbf{F}_t''(\mathbf{F}_t')^{-1}\mathbf{z} + \left(\min_{\mathbf{u}}\mathbf{u}^\top\mathbf{B}^\top\mathbf{F}_t\mathbf{B}\mathbf{u} - 2\mathbf{z}^\top\mathbf{B}\mathbf{u}\right) \\
&= \mathbf{z}^\top(\mathbf{F}_t')^{-1}\mathbf{F}_t''(\mathbf{F}_t')^{-1}\mathbf{z} + \left(\min_{\mathbf{s}=\mathbf{B}\mathbf{u}}\mathbf{s}^\top\mathbf{F}_t\mathbf{s} - 2\mathbf{z}^\top\mathbf{s}\right) \\
&= \min_{\mathbf{s}=\mathbf{B}\mathbf{u}}\mathbf{z}^\top(\mathbf{F}_t')^{-1}\mathbf{F}_t''(\mathbf{F}_t')^{-1}\mathbf{z} + \mathbf{s}^\top\mathbf{F}_t\mathbf{s} - 2\mathbf{z}^\top\mathbf{s} \\
&= \min_{\tilde{\mathbf{s}}}\tilde{\mathbf{z}}^\top\mathbf{F}_t\mathbf{F}_t''(\mathbf{F}_t')^{-2}\tilde{\mathbf{z}} + \tilde{\mathbf{s}}^\top\tilde{\mathbf{s}} - 2\tilde{\mathbf{z}}^\top\tilde{\mathbf{s}},\ \tilde{\mathbf{s}} = \mathbf{F}_t^{\frac{1}{2}}\mathbf{s},\ \tilde{\mathbf{z}} = \mathbf{F}_t^{-\frac{1}{2}}\mathbf{z}
\end{aligned}$$

Using equation E.1, which is equivalent to $\mathbf{F}_t\mathbf{F}_t''(\mathbf{F}_t')^{-2} \succeq \mathbf{I}$, we can further lower bound the above expression by

$$\mathbf{z}^\top(\mathbf{G}_t - \mathbf{S}_t)\mathbf{z} \geq \min_{\tilde{\mathbf{s}}}\tilde{\mathbf{z}}^\top\tilde{\mathbf{z}} + \tilde{\mathbf{s}}^\top\tilde{\mathbf{s}} - 2\tilde{\mathbf{z}}^\top\tilde{\mathbf{s}} \geq \min_{\tilde{\mathbf{s}}}\|\tilde{\mathbf{z}} - \tilde{\mathbf{s}}\|^2 \geq 0,$$

which completes the proof.

E.2 Concavity of log determinant

We will show that $\gamma \mapsto \mathbf{1}^\top\ln f(\gamma) + \ln|\mathbf{A}|$ with $\mathbf{A} = \mathbf{X}^\top\mathbf{X} + \mathbf{B}^\top[f(\mathbf{\Gamma})]^{-1}\mathbf{B}$ is concave whenever all scalar functions $f_j(\gamma_j) \geq 0$ are concave. by induction over the number of terms r in the sum $j = 1..q$.

$$\psi_r(\gamma) = \sum_{j=1}^{r}\ln f_j(\gamma_j) + \ln\left|\mathbf{X}^\top\mathbf{X} + \sum_{j=1}^{r}\mathbf{b}_j\mathbf{b}_j^\top\frac{1}{f_j(\gamma_j)}\right|$$

First of all, ln is a concave increasing function, therefore the concatenation $\ln f_j(\gamma_j)$ is concave [Boyd and Vandenberghe, 2004, equation 3.10]. Second, $\psi_0(\gamma)$ is constant and therefore concave. Now, supposing that $\psi_{r-1}(\gamma)$ is concave, we will show that $\psi_r(\gamma)$ is concave. We split

$$\begin{aligned}
\psi_r(\gamma) - \mathbf{1}^\top\ln f(\gamma) &= \sum_{j=1}^{r-1}\ln f_j(\gamma_j) + \ln f_r(\gamma_r) + \ln\left|\mathbf{X}^\top\mathbf{X} + \sum_{j=1}^{r-1}\mathbf{b}_j\mathbf{b}_j^\top\frac{1}{f_j(\gamma_j)} + \mathbf{b}_r\mathbf{b}_r^\top\frac{1}{f_r(\gamma_r)}\right| \\
&= f_{<r}(\gamma_{<r}) + \ln f_r(\gamma_r) + \ln\left|\mathbf{A}_{<r} + \mathbf{b}_r\mathbf{b}_r^\top\frac{1}{f_r(\gamma_r)}\right|
\end{aligned}$$

and rewrite using the matrix determinant lemma (appendix A.1.2) into

$$\begin{aligned}\psi_r(\gamma) &= f_{<r}(\gamma_{<r}) + \ln f_r(\gamma_r) + \ln|\mathbf{A}_{<r}| - \ln f_r(\gamma_r) + \ln\left(f_r(\gamma_r) + \mathbf{b}_r^\top \mathbf{A}_{<r}^{-1}\mathbf{b}_r\right) \\ &= \psi_{r-1}(\gamma) + \ln\left(f_r(\gamma_r) + \mathbf{b}_r^\top \mathbf{A}_{<r}^{-1}\mathbf{b}_r\right).\end{aligned}$$

Therefore, using monotonicity and concavity of the logarithm, we only need to show concavity of $\mathbf{b}_r^\top \mathbf{A}_{<r}^{-1}\mathbf{b}_r$ since $f_r(\gamma_r)$ is concave by assumption. Using Fenchel duality (appendix B.4), we can write

$$\begin{aligned}\frac{1}{2}\mathbf{b}_r^\top \mathbf{A}_{<r}^{-1}\mathbf{b}_r &= \max_{\mathbf{u}} \mathbf{b}_r^\top \mathbf{u} - \frac{1}{2}\mathbf{u}^\top \mathbf{A}_{<r}\mathbf{u} \\ &= \max_{\mathbf{u}} \mathbf{b}_r^\top \mathbf{u} - \frac{1}{2}\mathbf{u}^\top \mathbf{X}^\top \mathbf{X}\mathbf{u} - \frac{1}{2}\sum_{j=1}^{r-1}\frac{s_j^2}{f_j(\gamma_j)},\ \mathbf{s} = \mathbf{B}\mathbf{u}.\end{aligned}$$

Thus, the proof reduces to show that $s_j^2/\left(-f_j(\gamma_j)\right)$ is jointly convex in (s_j, γ_j) using Prékopa's theorem as in the simple approach in the previous section. The determinant of the Hessian $|\mathbf{H}_{s,\gamma}| = -2\frac{s^2}{f^2}\frac{f''}{f}$ is positive since $f \geq 0$ and $f'' \leq 0$, which completes the proof.

The case when $\mathbf{X}^\top\mathbf{X}$ is singular can be dealt with by starting from $\mathbf{X}^\top\mathbf{X} + \epsilon\mathbf{I}$ and considering the limit of $\epsilon \to 0$, which exists since all functions are assumed to be continuous.

E.3 Convexity of height functions

We focus on a single continuous symmetric potential $\mathcal{T}(s) \geq 0$ that is strongly super-Gaussian, i.e. $g(x) = \ln\mathcal{T}(s)$ is convex in $x := s^2 \geq 0$ and decreasing. We show that

- $h(\gamma)$ is convex if and only if $g(s) = \ln\mathcal{T}(s)$ is concave in s ($\mathcal{T}(s)$ is log-concave).

Using Fenchel duality [Rockafellar, 1970, chapter 12], we can represent $g(x)$ by the relationship $g(x) = \max_{p<0} xp - g^*(p)$ and hence $g(s) = \max_{\gamma>0} -\frac{1}{2\gamma}s^2 - g^*(-\frac{1}{2\gamma})$ substituting $x := s^2$ and $p := -\frac{1}{2\gamma} < 0$[1]. Note that the maximum is attained for $p = g'(x)$. As a result, we obtain a Gaussian lower bound on the potential [Palmer et al., 2006]

$$\mathcal{T}(s) = \max_{\gamma>0}\exp\left(-\frac{s^2}{2\gamma} - \frac{h(\gamma)}{2}\right),\ h(\gamma) = 2g^*\left(\frac{-1}{2\gamma}\right) \Leftrightarrow -2\ln\mathcal{T}(s) = \min_{\gamma>0}\frac{s^2}{\gamma} + h(\gamma). \quad \text{(E.2)}$$

"$\Leftarrow$": Now, for $h(\gamma)$ convex, the expression $s^2/\gamma + h(\gamma)$ in equation E.2 is jointly convex. Convexity is preserved under marginalisation (see appendix B.2) which implies concavity of $\ln\mathcal{T}(s)$ and concludes one direction of the equivalence.

"$\Rightarrow$": In turn, we can express $h(\gamma)$ by conjugate duality as

$$\begin{aligned}h(\gamma) &= \max_{s\geq 0} f(s,\gamma),\ f(s,\gamma) := -\frac{1}{\gamma}s^2 - 2g(s) \\ &= \max_{x\geq 0} f(x,\gamma),\ f(x,\gamma) := -\frac{1}{\gamma}x - 2g(x) \\ &= f(x_*(\gamma),\gamma),\ x_*(\gamma) := \arg\max_x f(x,\gamma).\end{aligned} \quad \text{(E.3)}$$

We obtain $x_*(\gamma)$ be setting $\frac{\partial}{\partial x}f(x,\gamma) = 0$ and solving for x yielding $g'(x_*) = -\frac{1}{2\gamma}$. Convexity of $g(x)$ implies $g''(x) \geq 0$ therefore invertibility of $g'(x)$, thus the relation $\gamma \mapsto x_*$ is unique.

[1] The values p are negative since the first derivative of $g(x)$ is negative, i.e. $g(x)$ is decreasing.

As a next step, we compute the derivative $\frac{\mathrm{d}}{\mathrm{d}\gamma}x_*(\gamma)$ by differentiating both sides of $g'(x_*) = -\frac{1}{2\gamma}$ w.r.t. γ, which leads to

$$\frac{\mathrm{d}}{\mathrm{d}\gamma}x_*(\gamma) \quad = \quad \frac{1}{2\gamma^2 g''(x_*)} \geq 0, \text{ since } g(x) \text{ is convex.}$$

From $x_*'(\gamma) \geq 0$ we deduce that $\gamma \mapsto x_*$ is increasing, which also holds for $\gamma \mapsto s_* = \sqrt{x_*}$ since the square root is increasing. Similarly, we compute $s_*(\gamma)$ by equating $\frac{\partial}{\partial s}f(s,\gamma)$ with 0, which gives $g'(s_*) = -\frac{1}{\gamma}s_* < 0$ and can be used to compute

$$h'(\gamma) = \frac{\partial}{\partial\gamma}f(s_*(\gamma),\gamma) = \frac{1}{\gamma^2}s_*^2 - \left(\overbrace{\frac{1}{\gamma}s_* + g'(s_*)}^{0}\right)\frac{\partial}{\partial\gamma}s_*(\gamma) = \frac{1}{\gamma^2}s_*^2 = \left[g'(s_*)\right]^2.$$

The concavity of $g(s)$ implies a monotonic decrease of $g'(s)$, which combined with $g'(s) < 0$ and the above derivation shows that the map $s_* \mapsto h'(\gamma)$ is an increasing one due to the square around $g'(s_*)$. Finally, we can conclude from the aforementioned fact that $\gamma \mapsto s_*$ is increasing that $\gamma \mapsto h'(\gamma)$ is monotonically increasing, too, and therefore $h(\gamma)$ is a convex function.

E.4 Generic inner loop computations

During the inner loop minimisation, the scalar functions defined by equation 3.11

$$h^*(s) \quad = \quad \frac{\sigma^2}{2}\min_{\gamma} h(s,\gamma),\ h(s,\gamma) = h_{\cup}(\gamma) + \left(\frac{s^2}{\sigma^2} + z_2\right)\gamma^{-1} + z_1\gamma - z_3\ln\gamma$$

that belong to the potential $\mathcal{T}(s)$ as well as their derivatives $h^{*\prime}(s)$ and $h^{*\prime\prime}(s)$ need to be evaluated for many different values of s. We dropped the subscript to simplify notation.

The minimiser $\gamma_* = \arg\min_\gamma h(s,\gamma)$ is found by standard convex optimisation techniques such as the Newton algorithm yielding $h^*(s) = h(s,\gamma_*)$. Using the total derivative and the fact $\frac{\partial}{\partial\gamma}h(s,\gamma_*) = 0$, we get

$$h^{*\prime}(s) = \frac{\mathrm{d}}{\mathrm{d}s}h(s,\gamma_*) = \frac{\sigma^2}{2}\left(\frac{\partial}{\partial s}h(s,\gamma_*) + \frac{\partial}{\partial\gamma}h(s,\gamma_*)\frac{\mathrm{d}\gamma_*}{\mathrm{d}s}\right) = \frac{\sigma^2}{2}\frac{\partial}{\partial s}h(s,\gamma_*) = \frac{s}{\gamma_*} =: \psi(s,\gamma_*),$$

where we defined the function $\psi(s,\gamma)$. Furthermore, the constraint

$$\frac{\partial}{\partial\gamma}h(s,\gamma_*) = h_\gamma(s,\gamma_*) = h_\cup'(\gamma_*) - \left(\frac{s^2}{\sigma^2} + z_2\right)\gamma_*^{-2} + z_1 - z_3\gamma_*^{-1} = 0$$

holds for all s, hence we have

$$\begin{aligned} \frac{\mathrm{d}}{\mathrm{d}s}h_\gamma(s,\gamma_*) = 0 \quad &= \quad \frac{\partial}{\partial s}h_\gamma(s,\gamma_*) + \frac{\partial}{\partial\gamma}h_\gamma(s,\gamma_*)\frac{\mathrm{d}\gamma_*}{\mathrm{d}s} \\ &= \quad -\frac{2s}{\sigma^2}\gamma_*^{-2} + \left[h_\cup''(\gamma_*) + 2\left(\frac{s^2}{\sigma^2} + z_2\right)\gamma_*^{-3} + z_3\gamma_*^{-2}\right]\frac{\mathrm{d}\gamma_*}{\mathrm{d}s} \\ \Leftrightarrow \frac{\mathrm{d}\gamma_*}{\mathrm{d}s} \quad &= \quad \frac{s\gamma_*}{s^2 + \gamma_*\kappa},\ \kappa = \sigma^2 z_2\gamma_*^{-1} + \frac{\sigma^2}{2}\gamma_*^2 h_\cup''(\gamma_*) + \frac{\sigma^2}{2}z_3. \end{aligned}$$

Using the total derivative once more, this can be combined into the second derivative

$$\begin{aligned} h^{*\prime\prime}(s) \quad &= \quad \frac{\mathrm{d}}{\mathrm{d}s}h^{*\prime}(s) = \frac{\mathrm{d}}{\mathrm{d}s}\psi(s,\gamma_*) = \left(\frac{\partial}{\partial s}\psi(s,\gamma_*) + \frac{\partial}{\partial\gamma}\psi(s,\gamma_*)\frac{\mathrm{d}\gamma_*}{\mathrm{d}s}\right) \\ &= \quad \frac{1}{\gamma_*} - \frac{s}{\gamma_*^2}\frac{\mathrm{d}\gamma_*}{\mathrm{d}s} = \frac{\kappa}{s^2 + \gamma_*\kappa}. \end{aligned}$$

We see, that in order to evaluate $h^*(s)$, $h^{*\prime}(s)$ and $h^{*\prime\prime}(s)$, the only things we need to compute from the potential $\mathcal{T}(s)$ is $h_\cup(\gamma)$ and $h''_\cup(\gamma)$. In the following, we show how these can be computed only from $g(x) = \ln \mathcal{T}(s)$, where $s^2 = x$ and its derivatives $g'(x)$ and $g''(x)$.

Starting from the definition of $h(\gamma)$ given in equation E.3, we can compute $h(\gamma)$ using a one-dimensional maximisation

$$h(\gamma) = \max_{x \geq 0} -f(x,\gamma),\ f(x,\gamma) = \frac{1}{\gamma}x + 2g(x).$$

All we need is the first and second derivative

$$\frac{\partial f}{\partial x} = \frac{1}{\gamma} + 2g'(x) \qquad \frac{\partial^2 f}{\partial x^2} = 2g''(x)$$

in order to apply Newton's method to compute x_* and hence $h(\gamma) = -f(x_*,\gamma)$. Using the same principles as above, we can compute

$$h'(\gamma) = -\frac{\partial}{\partial \gamma} f(x_*,\gamma) - \frac{\partial}{\partial x} f(x_*,\gamma)\frac{\mathrm{d}x_*}{\mathrm{d}\gamma} = -\frac{\partial}{\partial \gamma} f(x_*,\gamma) = \frac{1}{\gamma^2}x_* = -f_\gamma(x_*,\gamma)$$

and from the optimality condition $\frac{\partial}{\partial x} f(x_*,\gamma) = f_x(x_*,\gamma) = 0$ that holds and further from $\frac{\mathrm{d}}{\mathrm{d}\gamma} f_x(x_*,\gamma) = 0$, we get

$$\begin{aligned}
\frac{\mathrm{d}}{\mathrm{d}\gamma} f_x(x_*,\gamma) = 0 &= \frac{\partial}{\partial \gamma} f_x(x_*,\gamma) + \frac{\partial}{\partial x} f_x(x_*,\gamma)\frac{\mathrm{d}x_*}{\mathrm{d}\gamma} \\
&= -\frac{1}{\gamma^2} + 2g''(x_*)\frac{\mathrm{d}x_*}{\mathrm{d}\gamma} \\
\Leftrightarrow \frac{\mathrm{d}x_*}{\mathrm{d}\gamma} &= \frac{1}{2g''(x_*)\gamma^2}.
\end{aligned}$$

Hence, we can deduce

$$\begin{aligned}
h''(\gamma) &= -\frac{\mathrm{d}}{\mathrm{d}\gamma} f_\gamma(x_*,\gamma) = -\frac{\partial}{\partial \gamma} f_\gamma(x_*,\gamma) - \frac{\partial}{\partial x} f_\gamma(x_*,\gamma)\frac{\mathrm{d}x_*}{\mathrm{d}\gamma} \\
&= -\frac{2}{\gamma^3}x_* + \frac{1}{2g''(x_*)\gamma^4} = \gamma^{-4}\left(\frac{1}{2g''(x_*)} - 2x_*\gamma\right).
\end{aligned}$$

Assuming $h_\cup(\gamma) = h(\gamma) - h_\cap(\gamma)$ and $h''_\cap(\gamma)$ to be known, we can summarise

$$\begin{aligned}
\kappa &= \sigma^2 z_2 \gamma_*^{-1} + \frac{\sigma^2}{2}\gamma_*^2\left[h''(\gamma_*) - h''_\cap(\gamma_*)\right] + \frac{\sigma^2}{2}z_3 \\
&= \sigma^2 z_2 \gamma_*^{-1} + \frac{\sigma^2}{2}\gamma_*^{-2}\left(\frac{1}{2g''(x_*)} - 2x_*\gamma_*\right) - \frac{\sigma^2}{2}\gamma_*^2 h''_\cap(\gamma_*) + \frac{\sigma^2}{2}z_3 \\
&= \frac{\sigma^2}{2}\left(2(z_2 - x_*)\gamma_*^{-1} + \frac{1}{2\gamma_*^2 g''(x_*)} - \gamma_*^2 h''_\cap(\gamma_*) + z_3\right),
\end{aligned}$$

which can be used to compute $h^{*\prime\prime}(s) = \kappa/(s^2 + \gamma_*\kappa)$.

E.5 Generic inner loop for log-concave potentials

The result of appendix E.4 applies to any super Gaussian potential, however in the special case of log-concave potentials and the $\phi_\cup^{(2)}(\gamma)$ bound, where $z_1 = z_3 = 0$, the expressions for $h^*(s)$, $h^{*\prime}(s)$ and $h^{*\prime\prime}(s)$ become very simple. Using appendix E.3 and section 3.5.3, we have

$$h^*(s) = \sigma^2 \min_{\gamma \geq 0} \frac{1}{2}h(\gamma) + \frac{s^2 + \nu}{2\sigma^2\gamma} \quad \text{where} \quad \frac{1}{2}h(\gamma) = \max_{\tilde{s} \geq 0} -\frac{\tilde{s}^2}{2\sigma^2\gamma} + \frac{\beta}{\sigma^2}\tilde{s} - \ln \mathcal{T}(\tilde{s}),$$

which can be combined into a minimax expression for the penaliser of $h^*(s)$ as a function modulated by the marginal variances $\nu = \mathbb{V}_{\mathrm{Q}}[s|\mathcal{D}]$ of the posterior

$$h^*(s) = \sigma^2 \min_{\gamma \ge 0} \left(\max_{\tilde{s} \ge 0} \left(-\frac{\tilde{s}^2}{\gamma} + \frac{\beta}{\sigma^2}\tilde{s} - \ln \mathcal{T}_j(\tilde{s}) \right) + \frac{s^2+\nu}{\gamma} \right).$$

The inner expression $f_i(\tilde{s}) := -\tilde{s}^2/\gamma + \beta\tilde{s}/\sigma^2 - \ln \mathcal{T}(\tilde{s})$ is necessarily maximised for

$$0 = f_i'(\tilde{s}) = -2\frac{\tilde{s}}{\gamma} + \frac{\beta}{\sigma^2} - \frac{\mathcal{T}'(\tilde{s})}{\mathcal{T}(\tilde{s})} \quad \Leftrightarrow \quad 2\frac{\tilde{s}}{\gamma} = \frac{\beta}{\sigma^2} - \frac{\mathcal{T}'(\tilde{s})}{\mathcal{T}(\tilde{s})}$$

and its maximiser is denoted by $\tilde{s}^*_\gamma$ and is a function of γ. The remaining outer minimum of $f_o(\gamma) := (s^2 - (\tilde{s}^*_\gamma)^2 + \nu)/\gamma + \beta\tilde{s}^*_\gamma/\sigma^2 - \ln \mathcal{T}(\tilde{s}^*_\gamma)$ is attained if γ obeys

$$\begin{aligned} 0 = f_o'(\gamma) &= \frac{-2\tilde{s}^*_\gamma \frac{\mathrm{d}\tilde{s}^*_\gamma}{\mathrm{d}\gamma}\gamma - (s^2 - (\tilde{s}^*_\gamma)^2 + \nu)}{\gamma^2} + \left(\overbrace{\frac{\beta}{\sigma^2} - \frac{\mathcal{T}'(\tilde{s}^*_\gamma)}{\mathcal{T}(\tilde{s}^*_\gamma)}}^{2\frac{\tilde{s}}{\gamma}} \right) \frac{\mathrm{d}\tilde{s}^*_\gamma}{\mathrm{d}\gamma} \\ &= \frac{-(s^2 - (\tilde{s}^*_\gamma)^2 + \nu)}{\gamma^2} \Leftrightarrow \tilde{s}^*_\gamma = \mathrm{sign}(s)\sqrt{s^2+\nu}, \end{aligned}$$

where $\mathrm{sign}(x) \in \{\pm 1\}$ and where we used the conditions for the inner maximum in the derivation to finally obtain

$$h^*(s) = \sigma^2 \min_{\gamma \ge 0} \left(\frac{s^2 + \nu - (\tilde{s}^*_\gamma)^2}{\gamma} + \frac{\beta}{\sigma^2}\tilde{s}^*_\gamma - \ln \mathcal{T}(\tilde{s}^*_\gamma) \right) = \beta \mathrm{sign}(s)\sqrt{s^2+\nu} - \sigma^2 \ln \mathcal{T}\left(\mathrm{sign}(s)\sqrt{s^2+\nu} \right).$$

The derivatives of $h^*(s)$ are then very simple using $g(s) := \ln \mathcal{T}(s)$:

$$\begin{aligned} h^*(s) &= \beta\varsigma - \sigma^2 g(\varsigma), \; \varsigma = \mathrm{sign}(s)\sqrt{s^2+\nu} \\ h^{*\prime}(s) &= \left[\beta - \sigma^2 g'(\varsigma)\right] \frac{s}{\varsigma} \\ h^{*\prime\prime}(s) &= \left[\beta - \sigma^2 \left(g'(\varsigma) + \frac{s^2\varsigma}{\nu} g''(\varsigma) \right)\right] \frac{\nu}{\varsigma^3}. \end{aligned}$$

As a next step, we compute the minimum value of γ_* in $h^*(s)$. We start from the variational representation of a super-Gaussian potential (see equation 3.4 auf Seite 34)

$$\ln \mathcal{T}(s) - \frac{\beta s}{\sigma^2} = \max_\gamma \left[-\frac{s^2}{2\sigma^2\gamma} - \frac{1}{2}h(\gamma) \right]$$

and represent the equation using $x = s^2$, $p = -\frac{1}{2\sigma^2\gamma}$ and $g(s) = \ln \mathcal{T}(s)$

$$g(\sqrt{x}) - \frac{\beta\sqrt{x}}{\sigma^2} = \max_{p \le 0} \left[xp - \frac{1}{2}h(\gamma_{(p)}) \right],$$

where γ depends on p. Since the expression is in Legendre form (see appendix B.4), we know that the optimal value equals the derivative of the function on the left side

$$-\frac{1}{2\sigma^2\gamma_*} = p_* = \arg\max_{p \le 0} \left[xp - \frac{1}{2}h(\gamma_{(p)}) \right] = \frac{\mathrm{d}}{\mathrm{d}x}\left(g(\sqrt{x}) - \frac{\beta\sqrt{x}}{\sigma^2} \right).$$

We can simplify that expression and obtain

$$\begin{aligned} -\frac{1}{2\sigma^2\gamma_*} &= \frac{g'(\sqrt{x}) - \beta/\sigma^2}{2\sqrt{x}} = \frac{g'(s) - \beta/\sigma^2}{2|s|} \\ \Rightarrow \gamma_* &= \frac{\sqrt{x}}{\beta - \sigma^2 g'(\sqrt{x})}, \end{aligned}$$

where we have to set $x = s^2 + \nu = \varsigma^2$ as in $h^*(s)$ to finally obtain the inner loop update expression for γ

$$\gamma_* = \frac{\varsigma}{\beta - \sigma^2 g'(\varsigma)} = \frac{s}{h^{*\prime}(s)}.$$

E.6 SBL and variational bounds

We have two exact representations for symmetric super-Gaussian potentials (chapters 3.2 and 3.3)

$$\ln \mathcal{T}_j(s_j) = \begin{cases} \max_{\gamma_j} -\frac{1}{2}\left[\frac{s_j^2}{\sigma^2\gamma_j} + h(\gamma_j)\right] & \alpha)\text{ variational} \\ \ln \int \mathcal{N}(s_j|0,\sigma^2\gamma_j)\mathbb{P}_j(\gamma_j)\mathrm{d}\gamma_j = \ln \int \exp\left(-\frac{1}{2}\left[\frac{s_j^2}{\sigma^2\gamma_j} + p_j(\gamma_j)\right]\right)\mathrm{d}\gamma_j & \beta)\text{ scale mixture} \end{cases}$$

where $p_j(\gamma_j) = \ln(2\pi\sigma^2\gamma_j) - 2\ln\mathbb{P}_j(\gamma_j)$ and $\ln\mathcal{T}(\mathbf{s}) = \sum_j \ln\mathcal{T}_j(s_j)$. Three tools are used in the following:

- i) The maximum (variational) representation of the Gaussian partition function (equation 2.18 auf Seite 22)

$$\begin{aligned} \ln\tilde{Z}(\boldsymbol{\beta},\gamma) &= \ln\int \mathcal{N}(\mathbf{y}|\mathbf{X}\mathbf{u},\sigma^2\mathbf{I})\exp\left((\boldsymbol{\beta}^\top\mathbf{s} - \frac{1}{2}\mathbf{s}^\top\boldsymbol{\Gamma}^{-1}\mathbf{s})/\sigma^2\right)\mathrm{d}\mathbf{u} \\ &\overset{\text{c}}{=} \max_{\mathbf{u}} -\frac{1}{2}\left[R(\mathbf{u},\gamma)/\sigma^2 + \ln|\mathbf{A}|\right] \end{aligned}$$

  where $\mathbf{s} = \mathbf{B}\mathbf{u}$, $R(\mathbf{u},\gamma) = \|\mathbf{X}\mathbf{u}-\mathbf{y}\|^2 + \mathbf{s}^\top\boldsymbol{\Gamma}^{-1}\mathbf{s} - 2\boldsymbol{\beta}^\top\mathbf{s}$ and $\mathbf{A} = \mathbf{X}^\top\mathbf{X} + \mathbf{B}^\top\boldsymbol{\Gamma}^{-1}\mathbf{B}$,

- ii) the convex dual representation of the log determinant (equation 3.8 auf Seite 39)

$$-\frac{1}{2}\ln|\mathbf{A}| = \min_{\mathbf{z}} -\frac{1}{2}\left[\mathbf{z}^\top\gamma^{-1} - g^*(\mathbf{z})\right] \text{ and}$$

- iii) the inequality

$$\int \max_{\mathbf{x}} f(\mathbf{x},\mathbf{u})\mathrm{d}\mathbf{u} \geq \max_{\mathbf{x}} \int f(\mathbf{x},\mathbf{u})\mathrm{d}\mathbf{u}.$$

Starting from the two representations α) and β) and using the facts i-iii), we can derive to the same lower bound (equation 3.13 auf Seite 43 and appendix E.5 auf Seite 143) to the log partition function $\ln Z$:

$$\begin{aligned} \ln Z &\overset{\text{c}}{=} \ln\int \mathcal{N}(\mathbf{y}|\mathbf{X}\mathbf{u},\sigma^2\mathbf{I})\prod_j \mathcal{T}_j(s_j)\mathrm{d}\mathbf{u} \\ &\overset{\alpha)}{=} \ln\int \max_{\gamma}\mathcal{N}(\mathbf{y}|\mathbf{X}\mathbf{u},\sigma^2\mathbf{I})\exp\left(-\frac{1}{2}\left[\mathbf{s}^\top\boldsymbol{\Gamma}^{-1}\mathbf{s}/\sigma^2 + h(\gamma)\right]\right)\mathrm{d}\mathbf{u} \\ &\overset{\text{iii)}}{\geq} \max_{\gamma}\ln\int \mathcal{N}(\mathbf{y}|\mathbf{X}\mathbf{u},\sigma^2\mathbf{I})\exp\left(-\frac{1}{2}\left[(\mathbf{s}^2)^\top(\sigma^2\gamma)^{-1} + h(\gamma)\right]\right)\mathrm{d}\mathbf{u} \\ &\overset{\text{i)}}{=} \max_{\gamma,\mathbf{u}} -\frac{1}{2}\left[R(\mathbf{u},\gamma)/\sigma^2 + \ln|\mathbf{A}| + h(\gamma)\right] \\ &\overset{\text{ii)}}{=} \max_{\gamma,\mathbf{u},\mathbf{z}} -\frac{1}{2}\left[R(\mathbf{u},\gamma)/\sigma^2 + \mathbf{z}^\top\gamma^{-1} - g^*(\mathbf{z}) + h(\gamma)\right] \\ &= \max_{\gamma,\mathbf{u},\mathbf{z}} -\frac{1}{2}\left[\|\mathbf{X}\mathbf{u}-\mathbf{y}\|^2/\sigma^2 - g^*(\mathbf{z}) + (\mathbf{s}^2 + \sigma^2\mathbf{z})^\top(\sigma^2\gamma)^{-1} + h(\gamma)\right] \\ &\overset{\alpha)}{=} \max_{\mathbf{u},\mathbf{z}} -\frac{1}{2}\left[\|\mathbf{X}\mathbf{u}-\mathbf{y}\|^2/\sigma^2 - g^*(\mathbf{z}) - 2\ln\mathcal{T}(\sqrt{\mathbf{s}^2 + \sigma^2\mathbf{z}})\right] \end{aligned}$$

$$
\begin{aligned}
\ln Z &\overset{c}{=} \ln \int \mathcal{N}(\mathbf{y}|\mathbf{X}\mathbf{u}, \sigma^2\mathbf{I}) \prod_j \mathcal{T}_j(s_j) \mathrm{d}\mathbf{u} \\
&\overset{\beta)}{=} \ln \int \int \mathcal{N}(\mathbf{y}|\mathbf{X}\mathbf{u}, \sigma^2\mathbf{I}) \exp\left(-\frac{1}{2}\left[\mathbf{s}^\top \boldsymbol{\Gamma}^{-1}\mathbf{s}/\sigma^2 + p(\gamma)\right]\right) \mathrm{d}\mathbf{u}\mathrm{d}\boldsymbol{\gamma} \\
&\overset{i)}{=} \ln \int \exp\left(-\frac{1}{2}\left[\min_{\mathbf{u}} R(\mathbf{u},\gamma)/\sigma^2 + \ln|\mathbf{A}| + p(\gamma)\right]\right) \mathrm{d}\boldsymbol{\gamma} \\
&\overset{iii)}{\geq} \max_{\mathbf{u}} \ln \int \exp\left(-\frac{1}{2}\left[R(\mathbf{u},\gamma)/\sigma^2 + \ln|\mathbf{A}| + p(\gamma)\right]\right) \mathrm{d}\boldsymbol{\gamma} \\
&\overset{ii)}{=} \max_{\mathbf{u}} \ln \int \max_{\mathbf{z}} \exp\left(-\frac{1}{2}\left[R(\mathbf{u},\gamma)/\sigma^2 + \mathbf{z}^\top\gamma^{-1} - g^*(\mathbf{z}) + p(\gamma)\right]\right) \mathrm{d}\boldsymbol{\gamma} \\
&\overset{iii)}{\geq} \max_{\mathbf{u},\mathbf{z}} \ln \int \exp\left(-\frac{1}{2}\left[R(\mathbf{u},\gamma)/\sigma^2 + \mathbf{z}^\top\gamma^{-1} - g^*(\mathbf{z}) + p(\gamma)\right]\right) \mathrm{d}\boldsymbol{\gamma} \\
&= \max_{\mathbf{u},\mathbf{z}} -\frac{1}{2}\left[\left\|\mathbf{X}\mathbf{u} - \mathbf{y}\right\|^2/\sigma^2 - g^*(\mathbf{z}) - 2\ln\int \exp\left(-\frac{1}{2}\left[(\mathbf{s}^2 + \sigma^2\mathbf{z})^\top(\sigma^2\gamma)^{-1} + p(\gamma)\right]\right)\mathrm{d}\boldsymbol{\gamma}\right] \\
&\overset{\beta)}{=} \max_{\mathbf{u},\mathbf{z}} -\frac{1}{2}\left[\left\|\mathbf{X}\mathbf{u} - \mathbf{y}\right\|^2/\sigma^2 - g^*(\mathbf{z}) - 2\ln\mathcal{T}(\sqrt{\mathbf{s}^2 + \sigma^2\mathbf{z}})\right]
\end{aligned}
$$

Appendix F

Gaussian Process Classification

F.1 Derivatives for VB with ς-parametrisation

We start by some notational remarks. Partial derivatives w.r.t. one single parameter such as $\frac{\partial \mathbf{A}_\varsigma}{\partial \varsigma_i}$ or $\frac{\partial \mathbf{b}_\varsigma}{\partial \varsigma_i}$ stay matrices or vectors, respectively. Lowercase letters $\{\mathbf{a}, \mathbf{b}, \mathbf{c}\}_\varsigma$ indicate vectors, upper case letters $\{\mathbf{A}, \mathbf{B}, \mathbf{C}\}_\varsigma$ stand for the corresponding diagonal matrices with the vector as diagonal. The dot notation applies to both lower and uppercase letters and denote derivatives w.r.t. the variational parameter vector $\boldsymbol{\varsigma}$.

$$\begin{aligned}
\dot{\mathbf{a}}_\varsigma &:= \left[\frac{\partial a_{\varsigma_i}}{\partial \varsigma_i}\right]_i = \frac{\partial \mathbf{a}_\varsigma}{\partial \boldsymbol{\varsigma}}, \text{ vector} \\
\ddot{\mathbf{a}}_\varsigma &:= \left[\frac{\partial^2 a_{\varsigma_i}}{\partial \varsigma_i^2}\right]_i = \frac{\partial^2 \mathbf{a}_\varsigma}{\partial \boldsymbol{\varsigma}^2}, \text{ vector} \\
\dot{\mathbf{A}}_\varsigma &:= \mathrm{Dg}\left(\dot{\mathbf{a}}_\varsigma\right)
\end{aligned}$$

The operators $\mathrm{Dg} : \mathbb{R}^n \to \mathbb{R}^{n \times n}$ and $\mathrm{dg} : \mathbb{R}^{n \times n} \to \mathbb{R}^n$ manipulate matrix diagonals. The result of $\mathrm{Dg}(\mathbf{x})$ is a diagonal matrix $\mathbf{X}$ containing $\mathbf{x}$ as diagonal, whereas $\mathrm{dg}(\mathbf{X})$ returns the diagonal of $\mathbf{X}$ as a vector. Hence, we have $\mathrm{Dg}\left(\mathrm{dg}(\mathbf{x})\right) = \mathbf{x}$, but in general $\mathrm{dg}\left(\mathrm{Dg}(\mathbf{X})\right) = \mathbf{X}$ does only hold true for diagonal matrices.

F.1.0.1 Some shortcuts used later onwards:

$$\begin{aligned}
\tilde{\mathbf{K}}_\varsigma &:= \left(\mathbf{K}^{-1} - 2\mathbf{A}_\varsigma\right)^{-1} \overset{\mathrm{cond}\mathbf{K}\text{ small}}{=} \mathbf{K} - \mathbf{K}\left(\mathbf{K} - \frac{1}{2}\mathbf{A}_\varsigma^{-1}\right)^{-1}\mathbf{K} \\
\tilde{\mathbf{b}}_\varsigma &:= \mathrm{Dg}(\mathbf{y})\mathbf{b}_\varsigma = \mathbf{y} \odot \mathbf{b}_\varsigma \\
\mathbf{l}_\varsigma &:= \tilde{\mathbf{K}}_\varsigma \tilde{\mathbf{b}}_\varsigma = \left(\mathbf{K}^{-1} - 2\mathbf{A}_\varsigma\right)^{-1}\left(\mathbf{y} \odot \mathbf{b}_\varsigma\right) \\
\frac{\partial \mathbf{l}_\varsigma}{\partial \varsigma_j} &= \tilde{\mathbf{K}}_\varsigma \left(2\frac{\partial \mathbf{A}_\varsigma}{\partial \varsigma_j}\mathbf{l}_\varsigma + \mathbf{y} \odot \frac{\partial \mathbf{b}_\varsigma}{\partial \varsigma_j}\right) \\
\frac{\partial \mathbf{l}_\varsigma}{\partial \theta_i} &= \tilde{\mathbf{K}}_\varsigma \mathbf{K}^{-1}\frac{\partial \mathbf{K}}{\partial \theta_i}\mathbf{K}^{-1}\tilde{\mathbf{K}}_\varsigma\left(\mathbf{y} \odot \mathbf{b}_\varsigma\right) \\
\dot{\mathbf{L}}_\varsigma &:= \frac{\partial \mathbf{l}_\varsigma}{\partial \boldsymbol{\varsigma}^\top} = \tilde{\mathbf{K}}_\varsigma\left(2\mathrm{Dg}(\mathbf{l}_\varsigma)\dot{\mathbf{A}}_\varsigma + \mathrm{Dg}(\mathbf{y})\dot{\mathbf{B}}_\varsigma\right)
\end{aligned}$$

$$\begin{aligned}
\mathbf{r}_\varsigma &:= \dot{\mathbf{b}}_\varsigma \odot \mathbf{y} \odot \mathbf{l}_\varsigma + \mathrm{dg}\left(\mathbf{l}_\varsigma \mathbf{l}_\varsigma^\top \dot{\mathbf{A}}_\varsigma\right) \\
&= \dot{\mathbf{b}}_\varsigma \odot \mathbf{y} \odot \mathbf{l}_\varsigma + \mathbf{l}_\varsigma \odot \mathbf{l}_\varsigma \odot \dot{\mathbf{a}}_\varsigma \\
\frac{\partial \mathbf{r}_\varsigma}{\partial \varsigma_j} &= \mathbf{y} \odot \mathbf{l}_\varsigma \odot \frac{\partial \dot{\mathbf{b}}_\varsigma}{\partial \varsigma_j} + \dot{\mathbf{b}}_\varsigma \odot \mathbf{y} \odot \frac{\partial \mathbf{l}_\varsigma}{\partial \varsigma_j} + 2\mathbf{l}_\varsigma \odot \dot{\mathbf{a}}_\varsigma \odot \frac{\partial \mathbf{l}_\varsigma}{\partial \varsigma_j} + \mathbf{l}_\varsigma \odot \mathbf{l}_\varsigma \odot \frac{\partial \dot{\mathbf{a}}_\varsigma}{\partial \varsigma_j} \\
\dot{\mathbf{R}}_\varsigma &:= \frac{\partial \mathbf{r}_\varsigma}{\partial \boldsymbol{\varsigma}^\top} = \mathrm{Dg}\left(\mathbf{y} \odot \dot{\mathbf{b}}_\varsigma + 2\mathbf{l}_\varsigma \odot \dot{\mathbf{a}}_\varsigma\right) \dot{\mathbf{L}}_\varsigma + \mathrm{Dg}\left(\mathbf{l}_\varsigma \odot \left(\mathbf{y} \odot \ddot{\mathbf{b}}_\varsigma + \mathbf{l}_\varsigma \odot \ddot{\mathbf{a}}_\varsigma\right)\right) \\
&= \mathrm{Dg}\left(\mathbf{y} \odot \dot{\mathbf{b}}_\varsigma + 2\mathbf{l}_\varsigma \odot \dot{\mathbf{a}}_\varsigma\right) \tilde{\mathbf{K}}_\varsigma \mathrm{Dg}\left(\mathbf{y} \odot \dot{\mathbf{b}}_\varsigma + 2\mathbf{l}_\varsigma \odot \dot{\mathbf{a}}_\varsigma\right) + \mathrm{Dg}\left(\mathbf{l}_\varsigma \odot \left(\mathbf{y} \odot \ddot{\mathbf{b}}_\varsigma + \mathbf{l}_\varsigma \odot \ddot{\mathbf{a}}_\varsigma\right)\right)
\end{aligned}$$

F.1.0.2 First derivatives w.r.t. variational parameters ς_i yielding the gradient

$$\ln Z_{VB} = \mathbf{c}_\varsigma^\top \mathbf{1} + \frac{1}{2}\tilde{\mathbf{b}}_\varsigma^\top \tilde{\mathbf{K}}_\varsigma \tilde{\mathbf{b}}_\varsigma - \frac{1}{2}\ln\left|\mathbf{I} - 2\mathbf{A}_\varsigma \mathbf{K}\right| \tag{F.1}$$

$$\begin{aligned}
\frac{\partial \ln Z_{VB}}{\partial \varsigma_i} &= \frac{\partial c_i}{\partial \varsigma_i} + \tilde{\mathbf{b}}_\varsigma^\top \tilde{\mathbf{K}}_\varsigma \left[\mathbf{y} \odot \frac{\partial \mathbf{b}_\varsigma}{\partial \varsigma_i} + \frac{\partial \mathbf{A}_\varsigma}{\partial \varsigma_i} \tilde{\mathbf{K}}_\varsigma \tilde{\mathbf{b}}_\varsigma\right] + \mathrm{tr}\left(\left(\mathbf{I} - 2\mathbf{A}_\varsigma \mathbf{K}\right)^{-\top} \mathbf{K} \frac{\partial \mathbf{A}_\varsigma}{\partial \varsigma_i}\right) \\
&\overset{\mathbf{l}_\varsigma, \tilde{\mathbf{K}}_\varsigma}{=} \frac{\partial c_i}{\partial \varsigma_i} + \mathbf{l}_\varsigma^\top \left[\mathbf{y} \odot \frac{\partial \mathbf{b}_\varsigma}{\partial \varsigma_i} + \frac{\partial \mathbf{A}_\varsigma}{\partial \varsigma_i} \mathbf{l}_\varsigma\right] + \mathrm{tr}\left(\tilde{\mathbf{K}}_\varsigma \frac{\partial \mathbf{A}_\varsigma}{\partial \varsigma_i}\right) \\
\frac{\partial \ln Z_{VB}}{\partial \boldsymbol{\varsigma}} &= \left[\frac{\partial c_i}{\partial \varsigma_i}\right]_i + \dot{\mathbf{b}}_\varsigma \odot \mathbf{y} \odot \left(\tilde{\mathbf{K}}_\varsigma \tilde{\mathbf{b}}_\varsigma\right) + \mathrm{dg}\left(\tilde{\mathbf{K}}_\varsigma \tilde{\mathbf{b}}_\varsigma \tilde{\mathbf{b}}_\varsigma^\top \tilde{\mathbf{K}}_\varsigma \dot{\mathbf{A}}_\varsigma\right) + \mathrm{dg}\left(\tilde{\mathbf{K}}_\varsigma \dot{\mathbf{A}}_\varsigma\right) \\
&\overset{\mathbf{l}_\varsigma}{=} \left[\frac{\partial c_i}{\partial \varsigma_i}\right]_i + \dot{\mathbf{b}}_\varsigma \odot \mathbf{y} \odot \mathbf{l}_\varsigma + \mathrm{dg}\left(\mathbf{l}_\varsigma \mathbf{l}_\varsigma^\top \dot{\mathbf{A}}_\varsigma\right) + \mathrm{dg}\left(\tilde{\mathbf{K}}_\varsigma \dot{\mathbf{A}}_\varsigma\right) \\
&\overset{\mathbf{r}_\varsigma}{=} \left[\frac{\partial c_i}{\partial \varsigma_i}\right]_i + \mathbf{r}_\varsigma + \mathrm{dg}\left(\tilde{\mathbf{K}}_\varsigma \dot{\mathbf{A}}_\varsigma\right) \\
&= \dot{\mathbf{c}}_\varsigma + \mathbf{l}_\varsigma \odot \left(\dot{\mathbf{b}}_\varsigma \odot \mathbf{y} + \mathbf{l}_\varsigma \odot \dot{\mathbf{a}}_\varsigma\right) + \mathrm{dg}\left(\tilde{\mathbf{K}}_\varsigma\right) \odot \dot{\mathbf{a}}_\varsigma
\end{aligned}$$

F.1.0.3 Second derivatives w.r.t. variational parameters ς_i yielding the Hessian

$$\begin{aligned}
\frac{\partial^2 \ln Z_{VB}}{\partial \varsigma_j \partial \varsigma_i} &= \frac{\partial^2 c_i}{\partial \varsigma_j \partial \varsigma_i} + \frac{\partial \mathbf{r}_{\varsigma,i}}{\partial \varsigma_j} + \mathrm{tr}\left(2\tilde{\mathbf{K}}_\varsigma \frac{\partial \mathbf{A}_\varsigma}{\partial \varsigma_j} \tilde{\mathbf{K}}_\varsigma \frac{\partial \mathbf{A}_\varsigma}{\partial \varsigma_i} + \tilde{\mathbf{K}}_\varsigma \frac{\partial^2 \mathbf{A}_\varsigma}{\partial \varsigma_j \partial \varsigma_i}\right) \\
\frac{\partial^2 \ln Z_{VB}}{\partial \boldsymbol{\varsigma} \partial \boldsymbol{\varsigma}^\top} &= \left[\frac{\partial^2 c_i}{\partial \varsigma_i^2}\right]_{ii} + \frac{\partial \mathbf{r}_\varsigma}{\partial \boldsymbol{\varsigma}^\top} + 2\left(\tilde{\mathbf{K}}_\varsigma \dot{\mathbf{A}}_\varsigma\right) \odot \left(\tilde{\mathbf{K}}_\varsigma \dot{\mathbf{A}}_\varsigma\right)^\top + \mathrm{Dg}\left(\mathrm{dg}(\tilde{\mathbf{K}}_\varsigma) \odot \ddot{\mathbf{a}}_\varsigma\right) \\
&= \ddot{\mathbf{C}}_\varsigma + \dot{\mathbf{R}}_\varsigma + 2\left(\tilde{\mathbf{K}}_\varsigma \dot{\mathbf{A}}_\varsigma\right) \odot \left(\tilde{\mathbf{K}}_\varsigma \dot{\mathbf{A}}_\varsigma\right)^\top + \mathrm{Dg}\left(\mathrm{dg}(\tilde{\mathbf{K}}_\varsigma) \odot \ddot{\mathbf{a}}_\varsigma\right)
\end{aligned}$$

F.1.0.4 Mixed derivatives w.r.t. hyper- θ_i and variational parameters ς_i

$$\begin{aligned}
\frac{\partial^2 \ln Z_{VB}}{\partial \theta_i \partial \boldsymbol{\varsigma}} &= \dot{\mathbf{a}}_\varsigma \odot \frac{\partial}{\partial \theta_i}\left(\mathbf{l}_\varsigma \odot \mathbf{l}_\varsigma + \mathrm{dg}\left(\tilde{\mathbf{K}}_\varsigma\right)\right) + \dot{\mathbf{b}}_\varsigma \odot \mathbf{y} \odot \frac{\partial \mathbf{l}_\varsigma}{\partial \theta_i} \\
&= \dot{\mathbf{a}}_\varsigma \odot \left(2\mathbf{l}_\varsigma \odot \frac{\partial \mathbf{l}_\varsigma}{\partial \theta_i} + \mathrm{dg}\left(\tilde{\mathbf{K}}_\varsigma \mathbf{K}^{-1} \frac{\partial \mathbf{K}}{\partial \theta_i} \mathbf{K}^{-1} \tilde{\mathbf{K}}_\varsigma\right)\right) + \dot{\mathbf{b}}_\varsigma \odot \mathbf{y} \odot \frac{\partial \mathbf{l}_\varsigma}{\partial \theta_i}
\end{aligned}$$

F.1.0.5 First derivatives w.r.t. hyperparameters θ_i

For a gradient optimisation with respect to $\boldsymbol{\theta}$, we need the gradient of the objective $\partial \ln Z_B / \partial \boldsymbol{\theta}$

$$\begin{aligned}
\frac{\partial \ln Z_{VB}}{\partial \theta_i} &= \frac{1}{2}\tilde{\mathbf{b}}_\varsigma^\top \tilde{\mathbf{K}}_\varsigma \mathbf{K}^{-1} \frac{\partial \mathbf{K}}{\partial \theta_i} \mathbf{K}^{-1} \tilde{\mathbf{K}}_\varsigma \tilde{\mathbf{b}}_\varsigma + \mathrm{tr}\left(\left(\mathbf{I} - 2\mathbf{A}_\varsigma \mathbf{K}\right)^{-\top} \mathbf{A}_\varsigma \frac{\partial \mathbf{K}}{\partial \theta_i}\right) \\
&\overset{\mathbf{l}_\varsigma}{=} \frac{1}{2}\mathbf{l}_\varsigma^\top \mathbf{K}^{-1} \frac{\partial \mathbf{K}}{\partial \theta_i} \mathbf{K}^{-1} \mathbf{l}_\varsigma + \mathrm{tr}\left(\left(\mathbf{I} - 2\mathbf{A}_\varsigma \mathbf{K}\right)^{-\top} \mathbf{A}_\varsigma \frac{\partial \mathbf{K}}{\partial \theta_i}\right).
\end{aligned}$$

F.2 Derivatives for VB with γ-parametrisation

We compute the partial derivatives $\frac{\partial\phi}{\partial\gamma}, \frac{\partial\phi}{\partial\boldsymbol{\theta}}$ of

$$\begin{aligned}\phi(\gamma,\boldsymbol{\theta}) := -\frac{1}{2}\ln Z_{VB} &= \ln|\mathbf{K}_{\boldsymbol{\theta}}+\boldsymbol{\Gamma}| - \ln|\boldsymbol{\Gamma}| + h(\gamma) - \boldsymbol{\beta}^\top\left(\mathbf{K}_{\boldsymbol{\theta}}^{-1}+\boldsymbol{\Gamma}^{-1}\right)^{-1}\boldsymbol{\beta}\\ &= \ln\left|\tilde{\mathbf{K}}_{\boldsymbol{\theta}}\right| - \ln|\boldsymbol{\Gamma}| + h(\gamma) - \boldsymbol{\beta}^\top\hat{\mathbf{K}}_{\boldsymbol{\theta}}\boldsymbol{\beta},\end{aligned}$$

where we assume that $\boldsymbol{\beta}$ does not depend on $\boldsymbol{\theta}$ and define $\tilde{\mathbf{K}}_{\boldsymbol{\theta}} = \mathbf{K}_{\boldsymbol{\theta}}+\boldsymbol{\Gamma}$, $\hat{\mathbf{K}}_{\boldsymbol{\theta}} = (\mathbf{K}_{\boldsymbol{\theta}}^{-1}+\boldsymbol{\Gamma}^{-1})^{-1}$ as well as the shorthands $\mathbf{v} := \hat{\mathbf{K}}_{\boldsymbol{\theta}}\boldsymbol{\beta}$, $\mathbf{V} = \mathrm{dg}(\mathbf{v})$ and $\mathbf{w} = \mathbf{K}_{\boldsymbol{\theta}}^{-1}\mathbf{v} = \mathbf{K}_{\boldsymbol{\theta}}^{-1}\hat{\mathbf{K}}_{\boldsymbol{\theta}}\boldsymbol{\beta}$.

$$\begin{aligned}\mathrm{d}\phi &= \mathrm{tr}\left(\tilde{\mathbf{K}}_{\boldsymbol{\theta}}^{-1}(\mathrm{d}\mathbf{K}_{\boldsymbol{\theta}}+\mathrm{d}\boldsymbol{\Gamma})\right) + \left[-\gamma^{-1}+\mathbf{h}'(\gamma)\right]^\top\mathrm{d}\gamma\\ &\quad -\boldsymbol{\beta}^\top\hat{\mathbf{K}}_{\boldsymbol{\theta}}\left(\boldsymbol{\Gamma}^{-1}\mathrm{d}\boldsymbol{\Gamma}\boldsymbol{\Gamma}^{-1}+\mathbf{K}_{\boldsymbol{\theta}}^{-1}\mathrm{d}\mathbf{K}_{\boldsymbol{\theta}}\mathbf{K}_{\boldsymbol{\theta}}^{-1}\right)\hat{\mathbf{K}}_{\boldsymbol{\theta}}\boldsymbol{\beta} - 2\mathrm{d}\boldsymbol{\beta}^\top\hat{\mathbf{K}}_{\boldsymbol{\theta}}\boldsymbol{\beta}\\ \frac{\partial\phi}{\partial\gamma} &= \mathrm{dg}\left(\tilde{\mathbf{K}}_{\boldsymbol{\theta}}^{-1}\right) - \gamma^{-1} + \mathbf{h}'(\gamma) - \left(\mathbf{v}\odot\gamma^{-1}\right)^2 - 2\mathbf{v}\odot\boldsymbol{\beta}'\\ \frac{\partial\phi}{\partial\theta_i} &= \mathrm{tr}\left(\left[\tilde{\mathbf{K}}_{\boldsymbol{\theta}}^{-1}-\mathbf{K}_{\boldsymbol{\theta}}^{-1}\mathbf{v}\mathbf{v}^\top\mathbf{K}_{\boldsymbol{\theta}}^{-1}\right]\frac{\partial\mathbf{K}_{\boldsymbol{\theta}}}{\partial\theta_i}\right) = \mathrm{tr}\left(\left[\tilde{\mathbf{K}}_{\boldsymbol{\theta}}^{-1}-\mathbf{w}\mathbf{w}^\top\right]\frac{\partial\mathbf{K}_{\boldsymbol{\theta}}}{\partial\theta_i}\right)\end{aligned}$$

Computing the Hessian $\frac{\partial^2\phi}{\partial\gamma\partial\gamma^\top}$ requires a bit more work

$$\begin{aligned}\mathrm{d}^2\phi &= \mathrm{tr}(\mathrm{d}\tilde{\mathbf{K}}_{\boldsymbol{\theta}}^{-1}\mathrm{d}\boldsymbol{\Gamma}+\tilde{\mathbf{K}}_{\boldsymbol{\theta}}^{-1}\overbrace{\mathrm{d}^2\boldsymbol{\Gamma}}^{\mathbf{0}}) + \overbrace{\left[\gamma^{-2}+\mathbf{h}''(\gamma)\right]^\top(\mathrm{d}\gamma)^2}^{\xi} - \overbrace{\mathrm{d}\left(\boldsymbol{\beta}^\top\hat{\mathbf{K}}_{\boldsymbol{\theta}}\boldsymbol{\Gamma}^{-1}\mathrm{d}\boldsymbol{\Gamma}\boldsymbol{\Gamma}^{-1}\hat{\mathbf{K}}_{\boldsymbol{\theta}}\boldsymbol{\beta}\right)}^{\rho} - 2\mathrm{d}(\mathbf{v}^\top\mathrm{d}\boldsymbol{\beta})\\ &= -\mathrm{dg}\left(\tilde{\mathbf{K}}_{\boldsymbol{\theta}}^{-1}\mathrm{d}\boldsymbol{\Gamma}\tilde{\mathbf{K}}_{\boldsymbol{\theta}}^{-1}\right)^\top\mathrm{d}\gamma + \xi - \rho - 2\mathrm{d}\boldsymbol{\beta}^\top\left(\mathrm{d}\hat{\mathbf{K}}_{\boldsymbol{\theta}}\boldsymbol{\beta}+\hat{\mathbf{K}}_{\boldsymbol{\theta}}\mathrm{d}\boldsymbol{\beta}\right) - 2\mathbf{v}^\top\mathrm{d}^2\boldsymbol{\beta}\\ &= -(\mathrm{d}\gamma)^\top\left(\tilde{\mathbf{K}}_{\boldsymbol{\theta}}^{-1}\odot\tilde{\mathbf{K}}_{\boldsymbol{\theta}}^{-1}\right)\mathrm{d}\gamma + \xi - \rho - 2\mathrm{d}\boldsymbol{\beta}^\top\left(\hat{\mathbf{K}}_{\boldsymbol{\theta}}\mathrm{d}\boldsymbol{\Gamma}\boldsymbol{\Gamma}^{-2}\hat{\mathbf{K}}_{\boldsymbol{\theta}}\boldsymbol{\beta}+\hat{\mathbf{K}}_{\boldsymbol{\theta}}\mathrm{d}\boldsymbol{\beta}\right) - 2\mathbf{v}^\top\mathrm{d}^2\boldsymbol{\beta}\\ &= -(\mathrm{d}\gamma)^\top\left(\tilde{\mathbf{K}}_{\boldsymbol{\theta}}^{-1}\odot\tilde{\mathbf{K}}_{\boldsymbol{\theta}}^{-1}\right)\mathrm{d}\gamma + \xi - \rho - 2\mathrm{d}\boldsymbol{\beta}^\top\hat{\mathbf{K}}_{\boldsymbol{\theta}}\left(\boldsymbol{\Gamma}^{-2}\mathbf{V}\mathrm{d}\gamma+\mathrm{d}\boldsymbol{\beta}\right) - 2\mathbf{v}^\top\mathrm{d}^2\boldsymbol{\beta}\end{aligned}$$

$$\begin{aligned}\frac{\partial^2\phi}{\partial\gamma\partial\gamma^\top} &= -\tilde{\mathbf{K}}_{\boldsymbol{\theta}}^{-1}\odot\tilde{\mathbf{K}}_{\boldsymbol{\theta}}^{-1} + \boldsymbol{\Gamma}^{-2} + \mathrm{dg}\left[\mathbf{h}''(\gamma)\right] - 2\mathbf{V}\boldsymbol{\Gamma}^{-2}\hat{\mathbf{K}}_{\boldsymbol{\theta}}\left(\boldsymbol{\Gamma}^{-2}\mathbf{V}+2\mathrm{diag}(\boldsymbol{\beta}')\right)\\ &\quad -2\hat{\mathbf{K}}_{\boldsymbol{\theta}}\odot(\boldsymbol{\beta}'\boldsymbol{\beta}'^\top) + 2\boldsymbol{\Gamma}^{-3}\mathbf{V}^2 - 2\mathbf{V}\mathrm{diag}(\boldsymbol{\beta}''),\end{aligned}$$

where we used the derivation

$$\begin{aligned}\rho &:= \boldsymbol{\beta}^\top\left[\mathrm{d}\left(\hat{\mathbf{K}}_{\boldsymbol{\theta}}\{\boldsymbol{\Gamma}^{-1}\mathrm{d}\boldsymbol{\Gamma}\boldsymbol{\Gamma}^{-1}\}\hat{\mathbf{K}}_{\boldsymbol{\theta}}\right)\right]\boldsymbol{\beta} + 2\boldsymbol{\beta}^\top\hat{\mathbf{K}}_{\boldsymbol{\theta}}\boldsymbol{\Gamma}^{-1}\mathrm{d}\boldsymbol{\Gamma}\boldsymbol{\Gamma}^{-1}\hat{\mathbf{K}}_{\boldsymbol{\theta}}\mathrm{d}\boldsymbol{\beta}\\ &= 2\boldsymbol{\beta}^\top\left[\hat{\mathbf{K}}_{\boldsymbol{\theta}}\boldsymbol{\Gamma}^{-2}\mathrm{d}\boldsymbol{\Gamma}\hat{\mathbf{K}}_{\boldsymbol{\theta}}\boldsymbol{\Gamma}^{-2}\mathrm{d}\boldsymbol{\Gamma}\hat{\mathbf{K}}_{\boldsymbol{\theta}} - \hat{\mathbf{K}}_{\boldsymbol{\theta}}\boldsymbol{\Gamma}^{-3}(\mathrm{d}\boldsymbol{\Gamma})^2\hat{\mathbf{K}}_{\boldsymbol{\theta}}\right]\boldsymbol{\beta} + 2\boldsymbol{\beta}^\top\hat{\mathbf{K}}_{\boldsymbol{\theta}}\boldsymbol{\Gamma}^{-1}\mathrm{d}\boldsymbol{\Gamma}\boldsymbol{\Gamma}^{-1}\hat{\mathbf{K}}_{\boldsymbol{\theta}}\mathrm{d}\boldsymbol{\Gamma}\boldsymbol{\beta}'\\ &= 2\boldsymbol{\beta}^\top\hat{\mathbf{K}}_{\boldsymbol{\theta}}\left[\mathrm{d}\boldsymbol{\Gamma}\boldsymbol{\Gamma}^{-2}\hat{\mathbf{K}}_{\boldsymbol{\theta}}\boldsymbol{\Gamma}^{-2}\mathrm{d}\boldsymbol{\Gamma} - \boldsymbol{\Gamma}^{-3}(\mathrm{d}\boldsymbol{\Gamma})^2\right]\hat{\mathbf{K}}_{\boldsymbol{\theta}}\boldsymbol{\beta} + 2\mathbf{v}^\top\mathrm{d}\boldsymbol{\Gamma}\boldsymbol{\Gamma}^{-2}\hat{\mathbf{K}}_{\boldsymbol{\theta}}\mathrm{diag}(\boldsymbol{\beta}')\mathrm{d}\gamma\\ &= 2\mathbf{v}^\top\mathrm{d}\boldsymbol{\Gamma}\boldsymbol{\Gamma}^{-2}\hat{\mathbf{K}}_{\boldsymbol{\theta}}\boldsymbol{\Gamma}^{-2}\mathrm{d}\boldsymbol{\Gamma}\mathbf{v} - 2\left[\gamma^{-3}\odot\mathbf{v}^2\right]^\top(\mathrm{d}\gamma)^2 + 2(\mathrm{d}\gamma)^\top\mathbf{V}\boldsymbol{\Gamma}^{-2}\hat{\mathbf{K}}_{\boldsymbol{\theta}}\mathrm{diag}(\boldsymbol{\beta}')\mathrm{d}\gamma\\ &= 2(\mathrm{d}\gamma)^\top\mathbf{V}\boldsymbol{\Gamma}^{-2}\hat{\mathbf{K}}_{\boldsymbol{\theta}}\left(\boldsymbol{\Gamma}^{-2}\mathbf{V}+\mathrm{diag}(\boldsymbol{\beta}')\right)\mathrm{d}\gamma - 2\left[\gamma^{-3}\odot\mathbf{v}^2\right]^\top(\mathrm{d}\gamma)^2.\end{aligned}$$

F.3 Derivatives for KL

The lower bound $\ln Z_B$ to the log marginal likelihood $\ln Z$ is given by equation 4.13 as

$$\ln Z \;\geq\; = \ln Z_B(\mathbf{m},\mathbf{V}) = a(\mathbf{y},\mathbf{m},\mathbf{V}) + \frac{1}{2}\ln\left|\mathbf{V}\mathbf{K}^{-1}\right| + \frac{n}{2} - \frac{1}{2}\mathbf{m}^\top\mathbf{K}^{-1}\mathbf{m} - \frac{1}{2}\mathrm{tr}\left(\mathbf{V}\mathbf{K}^{-1}\right),$$

where we used the shortcut $a(\mathbf{y},\mathbf{m},\mathbf{V}) = \sum_{i=1}^{n} \int \mathcal{N}(f_i|m_i,v_{ii}) \ln \text{sig}(y_i f_i) \mathrm{d}f_i$. As a first step, we calculate the first derivatives of $\ln Z_B$ with respect to the posterior moments $\mathbf{m}$ and $\mathbf{V}$ to derive necessary conditions for the optimum by equating them with zero.

$$\frac{\partial \ln Z_B}{\partial \mathbf{V}} = \frac{\partial a(\mathbf{y},\mathbf{m},\mathbf{V})}{\partial \mathbf{V}} + \frac{1}{2}\mathbf{V}^{-1} - \frac{1}{2}\mathbf{K}^{-1} \overset{!}{=} \mathbf{0} \quad \Rightarrow \quad \mathbf{V} = \left(\mathbf{K}^{-1} - 2\text{Dgdg}\frac{\partial a}{\partial \mathbf{V}}\right)^{-1}$$

$$\frac{\partial \ln Z_B}{\partial \mathbf{m}} = \frac{\partial a(\mathbf{y},\mathbf{m},\mathbf{V})}{\partial \mathbf{m}} - \mathbf{K}^{-1}\mathbf{m} \overset{!}{=} \mathbf{0} \quad \Rightarrow \quad \mathbf{m} = \mathbf{K}\frac{\partial a}{\partial \mathbf{m}}$$

These two expressions are plugged in the original expression for $\ln Z_B$ using $\mathbf{A} = (\mathbf{I} - 2\mathbf{K}\mathbf{\Lambda})^{-1}$ and $\mathbf{\Lambda} = \text{Dgdg}\frac{\partial a}{\partial \mathbf{V}}$ to yield

$$\ln Z_B(\boldsymbol{\alpha},\mathbf{\Lambda}) = a\left(\mathbf{y},\mathbf{K}\boldsymbol{\alpha},(\mathbf{K}^{-1} - 2\mathbf{\Lambda})^{-1}\right) + \frac{1}{2}\ln|\mathbf{A}| - \frac{1}{2}\text{tr}\mathbf{A} + \frac{n}{2} - \frac{1}{2}\boldsymbol{\alpha}^\top\mathbf{K}\boldsymbol{\alpha}.$$

Our algorithm uses the parameters $\boldsymbol{\alpha}$, $\mathbf{\Lambda}$, so we calculate first and second derivatives to implement Newton's method.

F.3.0.6 First derivatives w.r.t. parameters $\boldsymbol{\alpha}$, $\mathbf{\Lambda}$ yielding the gradient

$$\frac{\partial \ln Z_B}{\partial \boldsymbol{\lambda}} = \frac{\partial a}{\partial \boldsymbol{\lambda}} + \text{dg}(\mathbf{V}) - \text{dg}(\mathbf{V}\mathbf{A}^\top) \quad \text{and} \quad \frac{\partial \ln Z_B}{\partial \boldsymbol{\alpha}} = \frac{\partial a}{\partial \boldsymbol{\alpha}} - \mathbf{K}\boldsymbol{\alpha}$$

Only the terms containing derivatives of a need further attention, namely

$$\frac{\partial a}{\partial \boldsymbol{\alpha}} = \mathbf{K}\frac{\partial a}{\partial \mathbf{m}} \qquad \text{and}$$

$$\begin{aligned} \text{d}(\text{dg}\mathbf{V}) &= \text{dg}\left[\text{d}\left(\mathbf{K}^{-1} - 2\mathbf{\Lambda}\right)^{-1}\right] = 2\text{dg}\left[\mathbf{V}\,\text{d}\mathbf{\Lambda}\,\mathbf{V}\right] = 2\text{dg}\left[\sum_k \mathbf{v}_k\mathbf{v}_k^\top \text{d}\lambda_k\right] = 2\sum_k (\mathbf{v}_k \odot \mathbf{v}_k)\,\text{d}\lambda_k \\ &= 2(\mathbf{V}\odot\mathbf{V})\,\text{d}\boldsymbol{\lambda} \Rightarrow \frac{\partial \text{dg}\mathbf{V}}{\partial \boldsymbol{\lambda}^\top} = 2\mathbf{V}\odot\mathbf{V} \\ \frac{\partial a}{\partial \boldsymbol{\lambda}} &= 2(\mathbf{V}\odot\mathbf{V})\frac{\partial a(\mathbf{y},\mathbf{m},\mathbf{V})}{\partial \text{dg}\mathbf{V}}. \end{aligned}$$

As a last step, the derivatives w.r.t. $\mathbf{m}$ and the diagonal part of $\mathbf{V}$ yield

$$\begin{aligned} \frac{\partial a}{\partial m_i} &= \int \frac{\partial \mathcal{N}(f|m_i,v_{ii})}{\partial m_i} \ln \text{sig}(y_i f)\text{d}f = \int \frac{f - m_i}{v_{ii}} \mathcal{N}(f|m_i,v_{ii}) \ln \text{sig}(y_i f)\text{d}f \\ &= \frac{1}{\sqrt{v_{ii}}} \int f \cdot \mathcal{N}(f) \ln \text{sig}\left(\sqrt{v_{ii}} y_i f + m_i y_i\right)\text{d}f \\ \frac{\partial a}{\partial v_{ii}} &= \int \frac{\partial \mathcal{N}(f|m_i,v_{ii})}{\partial v_{ii}} \ln \text{sig}(y_i f)\text{d}f = \int \left(\frac{(f-m_i)^2}{v_{ii}^{\frac{3}{2}}} - \frac{1}{\sqrt{v_{ii}}}\right) \mathcal{N}(f|m_i,v_{ii}) \ln \text{sig}(y_i f)\text{d}f \\ &= \frac{1}{2v_{ii}} \int (f^2 - 1) \cdot \mathcal{N}(f) \ln \text{sig}\left(\sqrt{v_{ii}} y_i f + m_i y_i\right)\text{d}f. \end{aligned}$$

F.3.0.7 Second derivatives w.r.t. parameters $\boldsymbol{\alpha}$, $\mathbf{\Lambda}$ yielding the Hessian

Again, we proceed in two steps, calculating derivatives w.r.t. $\boldsymbol{\alpha}$ and $\mathbf{\Lambda}$ and by the chain rule compute those w.r.t. $\mathbf{m}$ and $\mathbf{V}$.

$$
\begin{aligned}
\frac{\partial^2 \ln Z_B}{\partial \boldsymbol{\alpha} \partial \boldsymbol{\alpha}^\top} &= \frac{\partial^2 a}{\partial \boldsymbol{\alpha} \partial \boldsymbol{\alpha}^\top} + \mathbf{K} = \frac{\partial}{\partial \boldsymbol{\alpha}} \left[\frac{\partial a}{\partial \mathbf{m}^\top} \frac{\partial \mathbf{m}}{\partial \boldsymbol{\alpha}^\top} \right] + \mathbf{K} = \frac{\partial}{\partial \boldsymbol{\alpha}} \left[\frac{\partial a}{\partial \mathbf{m}^\top} \mathbf{K} \right] + \mathbf{K} \\
&= \frac{\partial}{\partial \boldsymbol{\alpha}} \left[\frac{\partial a}{\partial \mathbf{m}^\top} \right] \mathbf{K} + \mathbf{K} = \frac{\partial \mathbf{m}^\top}{\partial \boldsymbol{\alpha}} \frac{\partial}{\partial \mathbf{m}} \left[\frac{\partial a}{\partial \mathbf{m}^\top} \right] \mathbf{K} + \mathbf{K} \\
&= \mathbf{K} \frac{\partial^2 a}{\partial \mathbf{m} \partial \mathbf{m}^\top} \mathbf{K} + \mathbf{K} \\
\frac{\partial^2 \ln Z_B}{\partial \boldsymbol{\lambda} \partial \boldsymbol{\alpha}^\top} &= \frac{\partial^2 a}{\partial \boldsymbol{\lambda} \partial \boldsymbol{\alpha}^\top} = \frac{\partial}{\partial \boldsymbol{\lambda}} \left[\frac{\partial a}{\partial \mathbf{m}^\top} \right] \mathbf{K} = \frac{\partial \left(\mathrm{dg} \mathbf{V} \right)^\top}{\partial \boldsymbol{\lambda}} \frac{\partial}{\partial \mathrm{dg} \mathbf{V}} \left[\frac{\partial a}{\partial \mathbf{m}^\top} \right] \mathbf{K} \\
&= 2 \mathbf{V} \odot \mathbf{V} \frac{\partial^2 a}{\partial \mathrm{dg} \mathbf{V} \partial \mathbf{m}^\top} \mathbf{K}
\end{aligned}
$$

$$
\begin{aligned}
\frac{\partial^2 \ln Z_B}{\partial \boldsymbol{\lambda} \partial \boldsymbol{\lambda}^\top} &= \frac{\partial^2 a}{\partial \boldsymbol{\lambda} \partial \boldsymbol{\lambda}^\top} + 2 \mathbf{V} \odot \left(\mathbf{V} - \mathbf{A} \mathbf{V}^\top - \mathbf{V} \mathbf{A}^\top \right) \\
&= 2 \frac{\partial}{\partial \boldsymbol{\lambda}} \left[\frac{\partial a}{\partial \left(\mathrm{dg} \mathbf{V} \right)^\top} \mathbf{V} \odot \mathbf{V} \right] + 2 \mathbf{V} \odot \left(\mathbf{V} - \mathbf{A} \mathbf{V}^\top - \mathbf{V} \mathbf{A}^\top \right) \\
&= 2 \frac{\partial^2 a}{\partial \boldsymbol{\lambda} \partial \left(\mathrm{dg} \mathbf{V} \right)^\top} \mathbf{V} \odot \mathbf{V} + 2 \left[\frac{\partial a}{\partial \left(\mathrm{dg} \mathbf{V} \right)^\top} \frac{\partial \mathbf{V} \odot \mathbf{V}}{\partial \lambda_i} \right]_i + \overbrace{2 \mathbf{V} \odot \left(\mathbf{V} - \mathbf{A} \mathbf{V}^\top - \mathbf{V} \mathbf{A}^\top \right)}^{\mathbf{H}} \\
&= 2 \frac{\partial \left(\mathrm{dg} \mathbf{V} \right)^\top}{\partial \boldsymbol{\lambda}} \frac{\partial^2 a}{\partial \mathrm{dg} \mathbf{V} \partial \left(\mathrm{dg} \mathbf{V} \right)^\top} \mathbf{V} \odot \mathbf{V} + 4 \left[\frac{\partial a}{\partial \left(\mathrm{dg} \mathbf{V} \right)^\top} \left(\mathbf{V} \odot \frac{\partial \mathbf{V}}{\partial \lambda_i} \right) \right]_i + \mathbf{H} \\
&= 4 \mathbf{V} \odot \mathbf{V} \frac{\partial^2 a}{\partial \mathrm{dg} \mathbf{V} \partial \left(\mathrm{dg} \mathbf{V} \right)^\top} \mathbf{V} \odot \mathbf{V} + 8 \left[\frac{\partial a}{\partial \left(\mathrm{dg} \mathbf{V} \right)^\top} \left(\mathbf{V} \odot \left(\mathbf{v}_i \mathbf{v}_i^\top \right) \right) \right]_i + \mathbf{H}
\end{aligned}
$$

$$
\begin{aligned}
\frac{\partial^2 a}{\partial m_i^2} &= \int \frac{\partial^2 \mathcal{N}(f|m_i, v_{ii})}{\partial m_i^2} \ln \mathrm{sig}(y_i f) \mathrm{d}f = \int \frac{(f - m_i)^2 - c_{ii}}{v_{ii}^2} \mathcal{N}(f|m_i, v_{ii}) \ln \mathrm{sig}(y_i f) \mathrm{d}f \\
&= \frac{1}{v_{ii}} \int \left(f^2 - 1 \right) \cdot \mathcal{N}(f) \ln \mathrm{sig}(\sqrt{v_{ii}} y_i f + m_i y_i) \mathrm{d}f \\
\frac{\partial^2 a}{\partial c_{ii} \partial m_i} &= \int \frac{\partial^2 \mathcal{N}(f|m_i, v_{ii})}{\partial v_{ii} \partial m_i} \ln \mathrm{sig}(y_i f) \mathrm{d}f \\
&= \int \frac{(f - m_i)^3 - 3(f - m_i) v_{ii}}{2 v_{ii}^3} \mathcal{N}(f|m_i, v_{ii}) \ln \mathrm{sig}(y_i f) \mathrm{d}f \\
&= \frac{1}{2 v_{ii}^{\frac{3}{2}}} \int \left(f^3 - 3f \right) \cdot \mathcal{N}(f) \ln \mathrm{sig} \left(\sqrt{v_{ii}} y_i f + m_i y_i \right) \mathrm{d}f \\
\frac{\partial^2 a}{\partial v_{ii}^2} &= \int \frac{\partial^2 \mathcal{N}(f|m_i, v_{ii})}{\partial v_{ii}^2} \ln \mathrm{sig}(y_i f) \mathrm{d}f \\
&= \int \frac{(f - m_i)^4 - 6 v_{ii} (f - m_i)^2 + 3 v_{ii}^2}{4 v_{ii}^4} \mathcal{N}(f|m_i, v_{ii}) \ln \mathrm{sig}(y_i f) \mathrm{d}f \\
&= \frac{1}{4 v_{ii}^2} \int \left(f^4 - 6 f^2 + 3 \right) \cdot \mathcal{N}(f) \ln \mathrm{sig}(\sqrt{v_{ii}} y_i f + m_i y_i) \mathrm{d}f
\end{aligned}
$$

F.3.0.8 First derivatives w.r.t. hyperparameters θ_i

The direct gradient is given by the following equation, where we marked the dependency of the covariance $\mathbf{K}$ on θ_i by subscripts

$$\begin{aligned}\frac{\partial \ln Z_B(\boldsymbol{\alpha}, \boldsymbol{\Lambda})}{\partial \theta_i} &= \boldsymbol{\alpha}^\top \frac{\partial \mathbf{K}_{\boldsymbol{\theta}}}{\partial \theta_i} \frac{\partial a(\mathbf{y}, \mathbf{m}, \mathbf{V})}{\partial \mathbf{m}} + \mathrm{dg}\left(\mathbf{A}\frac{\partial \mathbf{K}_{\boldsymbol{\theta}}}{\partial \theta_i}\mathbf{A}^\top\right)^\top \frac{\partial a(\mathbf{y}, \mathbf{m}, \mathbf{V})}{\partial \mathrm{dg}\mathbf{V}} \\ &+ \mathrm{tr}\left(\mathbf{A}^\top \boldsymbol{\Lambda} \frac{\partial \mathbf{K}_{\boldsymbol{\theta}}}{\partial \theta_i}\right) - \mathrm{tr}\left(\mathbf{A}\frac{\partial \mathbf{K}_{\boldsymbol{\theta}}}{\partial \theta_i}\boldsymbol{\Lambda}\mathbf{A}\right) - \frac{1}{2}\boldsymbol{\alpha}^\top \frac{\partial \mathbf{K}_{\boldsymbol{\theta}}}{\partial \theta_i}\boldsymbol{\alpha}.\end{aligned}$$

F.4 Limits of the covariance matrix and marginal likelihood

We investigate the behaviour of the covariance matrix $\mathbf{K}$ for extreme length scales ℓ. The matrix is given by $[\mathbf{K}]_{ij} = \sigma_f^2 g(|\mathbf{x}_i - \mathbf{x}_j|/\ell)$, where $g : \mathbb{R} \to \mathbb{R}$ is monotonously decreasing and continuous with $g(0) = 1$ and $\lim_{t\to\infty} g(t) = 0$. From this definition we have $[\mathbf{K}]_{ii} = \sigma_f^2$. We define $\Delta_{ij} := |\mathbf{x}_i - \mathbf{x}_j|/\ell > 0$ for $i \neq j$. From

$$\begin{aligned}\lim_{\ell\to 0}[\mathbf{K}]_{ij} &\overset{i\neq j}{=} \lim_{\ell\to 0}\sigma_f^2 g(|\mathbf{x}_i - \mathbf{x}_j|/\ell) = \sigma_f^2 \lim_{\Delta_{ij}\to\infty} g(\Delta_{ij}) = 0 \\ \lim_{\ell\to\infty}[\mathbf{K}]_{ij} &\overset{i\neq j}{=} \lim_{\ell\to\infty}\sigma_f^2 g(|\mathbf{x}_i - \mathbf{x}_j|/\ell) = \sigma_f^2 \lim_{\Delta_{ij}\to 0} g(\Delta_{ij}) = 1\end{aligned}$$

we conclude

$$\begin{aligned}\lim_{\ell\to 0}\mathbf{K} &= \sigma_f^2\mathbf{I} \\ \lim_{\ell\to\infty}\mathbf{K} &= \sigma_f^2\mathbf{1}\mathbf{1}^\top.\end{aligned}$$

The sigmoid transfer functions are normalised $\mathrm{sig}\,(-f_i) + \mathrm{sig}\,(f_i) = 1$ and the Gaussian is symmetric $\mathcal{N}(f_i) = \mathcal{N}(-f_i)$. Consequently, we have

$$\begin{aligned}\int \mathrm{sig}\,(y_i f_i)\,\mathcal{N}(f_i|0,\sigma_f^2)\mathrm{d}f_i &= \int \mathrm{sig}\,(f_i)\,\mathcal{N}(f_i|0,\sigma_f^2)\mathrm{d}f_i \\ &= \int_{-\infty}^{0} \mathrm{sig}\,(f_i)\,\mathcal{N}(f_i|0,\sigma_f^2)\mathrm{d}f_i + \int_0^\infty \mathrm{sig}\,(f_i)\,\mathcal{N}(f_i|0,\sigma_f^2)\mathrm{d}f_i \\ &= \int_0^\infty \mathrm{sig}\,(-f_i)\,\mathcal{N}(-f_i|0,\sigma_f^2)\mathrm{d}f_i + \int_0^\infty \mathrm{sig}\,(f_i)\,\mathcal{N}(f_i|0,\sigma_f^2)\mathrm{d}f_i \\ &= \int_0^\infty [\mathrm{sig}\,(-f_i) + \mathrm{sig}\,(f_i)]\,\mathcal{N}(f_i|0,\sigma_f^2)\mathrm{d}f_i \\ &= \int_0^\infty 1\cdot\mathcal{N}(f_i|0,\sigma_f^2)\mathrm{d}f_i = \frac{1}{2} \qquad \text{(F.2)}\end{aligned}$$

The marginal likelihood is given by

$$\begin{aligned}Z &= \int \mathbb{P}\,(\mathbf{y}|\mathbf{f})\,\mathbb{P}\,(\mathbf{f}|\mathbf{X},\boldsymbol{\theta})\,\mathrm{d}\mathbf{f} \\ &= \int \prod_{i=1}^{n} \mathrm{sig}\,(y_i f_i)\,|2\pi\mathbf{K}|^{-\frac{1}{2}} \exp(-\frac{1}{2}\mathbf{f}^\top\mathbf{K}^{-1}\mathbf{f})\mathrm{d}\mathbf{f}.\end{aligned}$$

F.4.0.9 Length scale to zero

For $\mathbf{K} = \sigma_f^2 \mathbf{I}$ the prior factorises and we get

$$\begin{aligned} Z_{\ell \to 0} &= \prod_{i=1}^{n} \int \mathrm{sig}\,(y_i f_i) \frac{1}{\sqrt{2\pi\sigma_f^2}} \exp(-\frac{f_i^2}{2\sigma_f^2}) \mathrm{d}f_i \\ &\overset{(F.2)}{=} \prod_{i=1}^{n} \frac{1}{2} = 2^{-n}. \end{aligned}$$

F.4.0.10 Length scale to infinity

To get $\mathbf{K} \to \sigma_f^2 \mathbf{1}\mathbf{1}^\top$ we write $\mathbf{K} = \sigma_f^2 \mathbf{1} + \epsilon^2 \mathbf{I}$ with $\mathbf{1} = \mathbf{1}\mathbf{1}^\top$ and let $\epsilon \to 0$. The eigenvalue decomposition of $\mathbf{K}$ is written as $\mathbf{K} = \sum_{i=1}^n \mathbf{u}_i \mathbf{u}_i^\top \lambda_i$ with $\mathbf{u}_1 = \frac{1}{\sqrt{n}} \mathbf{1}$, $\lambda_1 = \sigma_f^2 + \epsilon^2$ and all other $\lambda_i = \epsilon^2$.

$$\begin{aligned} Z_{\frac{1}{\epsilon}} &\overset{\mathbf{K}=\mathbf{U}\mathbf{\Lambda}\mathbf{U}^\top}{=} \int \prod_{i=1}^{n} \mathrm{sig}\,(y_i f_i) \left|2\pi\mathbf{\Lambda}\right|^{-\frac{1}{2}} \exp(-\frac{1}{2}\mathbf{f}^\top \mathbf{U}\mathbf{\Lambda}^{-1}\mathbf{U}^\top \mathbf{f}) \mathrm{d}\mathbf{f} \\ &\overset{\mathbf{t}=\mathbf{\Lambda}^{-\frac{1}{2}}\mathbf{U}^\top\mathbf{f}}{=} \int \prod_{i=1}^{n} \mathrm{sig}\left(y_i \sqrt{\lambda_i} \cdot \mathbf{t}^\top \mathbf{u}_i\right) \left|2\pi\mathbf{\Lambda}\right|^{-\frac{1}{2}} \exp(-\frac{1}{2}\mathbf{t}^\top\mathbf{t}) \left|\mathbf{\Lambda}^{\frac{1}{2}}\right| \mathrm{d}\mathbf{t} \\ &= \int \prod_{i=1}^{n} \mathrm{sig}\left(y_i \sqrt{\lambda_i} \cdot \mathbf{t}^\top \mathbf{u}_i\right) \mathcal{N}(t_i) \mathrm{d}\mathbf{t} \\ &= \int \mathrm{sig}\left(\sqrt{\frac{\sigma_f^2 + \epsilon^2}{n}} \cdot \mathbf{t}^\top \mathbf{1}\right) \mathcal{N}(t_1) \prod_{i=2}^{n} \left[\mathrm{sig}\left(\epsilon \cdot \mathbf{t}^\top \mathbf{u}_i\right)\right] \mathcal{N}(t_i) \mathrm{d}\mathbf{t} \end{aligned}$$

$$\begin{aligned} Z_{\ell \to \infty} = \lim_{\epsilon \to 0} Z &= \int \mathrm{sig}\left(\frac{\sigma_f}{\sqrt{n}} \cdot \mathbf{t}^\top \mathbf{1}\right) \mathcal{N}(t_1) \prod_{i=2}^{n} \left[\frac{1}{2}\right] \mathcal{N}(t_i) \mathrm{d}\mathbf{t} \\ &\overset{(F.2)}{=} 2^{-n+1} \int \mathrm{sig}\left(\frac{\sigma_f}{\sqrt{n}} \cdot \mathbf{t}^\top \mathbf{1}\right) \mathcal{N}(\mathbf{t}) \mathrm{d}\mathbf{t} \\ &\overset{r=\mathbf{t}^\top\mathbf{1}}{=} 2^{-n+1} \int \mathrm{sig}\left(\frac{\sigma_f}{\sqrt{n}} \cdot r\right) \mathcal{N}(r) \mathrm{d}r \\ &\overset{(F.2)}{=} 2^{-n}. \end{aligned}$$

F.4.0.11 Latent scale to zero

We define $\sigma_f^2 \tilde{\mathbf{K}} = \mathbf{K}$ and $\sigma_f \tilde{\mathbf{f}} = \mathbf{f}$ and derive

$$\begin{aligned} Z_{\sigma_f} &= \int \prod_{i=1}^{n} \mathrm{sig}\,(y_i f_i) \left|2\pi\mathbf{K}\right|^{-\frac{1}{2}} \exp(-\frac{1}{2}\mathbf{f}^\top \mathbf{K}^{-1}\mathbf{f}) \mathrm{d}\mathbf{f} \\ &= \int \prod_{i=1}^{n} \mathrm{sig}\,(y_i \sigma_f \tilde{f}_i) \left|2\pi\mathbf{K}\right|^{-\frac{1}{2}} \exp(-\frac{\sigma_f^2}{2}\tilde{\mathbf{f}}^\top \mathbf{K}^{-1}\tilde{\mathbf{f}}) \sigma_f^n \mathrm{d}\tilde{\mathbf{f}} \\ &= \int \prod_{i=1}^{n} \mathrm{sig}\,(y_i \sigma_f \tilde{f}_i) \left|2\pi\sigma_f^2\tilde{\mathbf{K}}\right|^{-\frac{1}{2}} \exp(-\frac{\sigma_f^2}{2}\tilde{\mathbf{f}}^\top \sigma_f^{-2}\tilde{\mathbf{K}}^{-1}\tilde{\mathbf{f}}) \sigma_f^n \mathrm{d}\tilde{\mathbf{f}} \\ &= \int \prod_{i=1}^{n} \left[\mathrm{sig}\,(y_i \sigma_f \tilde{f}_i)\right] \mathcal{N}\left(\tilde{\mathbf{f}}|\mathbf{0}, \tilde{\mathbf{K}}\right) \mathrm{d}\tilde{\mathbf{f}} \\ Z_{\sigma_f \to 0} = \lim_{\sigma_f \to 0} Z &= \int \prod_{i=1}^{n} \left[\frac{1}{2}\right] \mathcal{N}\left(\tilde{\mathbf{f}}|\mathbf{0}, \tilde{\mathbf{K}}\right) \mathrm{d}\tilde{\mathbf{f}} = 2^{-n}. \end{aligned}$$

Note that the functions, we are using are all well-behaved, so that the limits do exist.

F.5 Posterior divided by prior = effective likelihood

$$\begin{aligned}
\mathbb{Q}(\mathbf{y}|\mathbf{f}) &= \frac{\mathcal{N}(\mathbf{f}|\mathbf{m},\mathbf{V})}{\mathbb{P}(\mathbf{f}|\mathbf{X})} = \frac{\mathcal{N}\left(\mathbf{f}|\mathbf{m},(\mathbf{K}^{-1}+\mathbf{W})^{-1}\right)}{\mathcal{N}(\mathbf{f}|\mathbf{0},\mathbf{K})}\\
&= \frac{\mathcal{N}(\mathbf{f}|\tilde{\mathbf{m}},\mathbf{W}^{-1})}{\mathcal{N}(\tilde{\mathbf{m}}|\mathbf{0},\mathbf{K}+\mathbf{W}^{-1})}, \quad \tilde{\mathbf{m}} = (\mathbf{KW})^{-1}\mathbf{m}+\mathbf{m}\\
&= \frac{(2\pi)^{-\frac{n}{2}}\left|\mathbf{W}^{-1}\right|^{-\frac{1}{2}}\exp\left(-\frac{1}{2}(\mathbf{f}-\tilde{\mathbf{m}})^\top\mathbf{W}(\mathbf{f}-\tilde{\mathbf{m}})\right)}{(2\pi)^{-\frac{n}{2}}\left|\mathbf{K}+\mathbf{W}^{-1}\right|^{-\frac{1}{2}}\exp\left(-\frac{1}{2}\tilde{\mathbf{m}}^\top(\mathbf{K}+\mathbf{W}^{-1})^{-1}\tilde{\mathbf{m}}\right)}\\
&= \sqrt{|\mathbf{KW}+\mathbf{I}|}\frac{\exp\left(-\frac{1}{2}(\mathbf{f}-\tilde{\mathbf{m}})^\top\mathbf{W}(\mathbf{f}-\tilde{\mathbf{m}})\right)}{\exp\left(-\frac{1}{2}\tilde{\mathbf{m}}^\top(\mathbf{K}+\mathbf{W}^{-1})^{-1}\tilde{\mathbf{m}}\right)}\\
&=: \frac{1}{Z_{\mathbb{Q}}}\exp\left(-\frac{1}{2}(\mathbf{f}-\tilde{\mathbf{m}})^\top\mathbf{W}(\mathbf{f}-\tilde{\mathbf{m}})\right)\\
\ln Z_{\mathbb{Q}} &= -\frac{1}{2}\tilde{\mathbf{m}}^\top\left(\mathbf{K}+\mathbf{W}^{-1}\right)^{-1}\tilde{\mathbf{m}} - \frac{1}{2}\ln|\mathbf{KW}+\mathbf{I}|
\end{aligned}$$

F.6 Kullback-Leibler divergence for KL method

We wish to calculate the divergence between the approximate posterior, a Gaussian, and the true posterior

$$\begin{aligned}
\mathrm{KL}\left(\mathbb{Q}(\mathbf{f}|\boldsymbol{\theta}) \;||\; \mathbb{P}(\mathbf{f}|\mathbf{y},\mathbf{X},\boldsymbol{\theta})\right) &= \int\mathcal{N}(\mathbf{f}|\mathbf{m},\mathbf{V})\ln\frac{\mathcal{N}(\mathbf{f}|\mathbf{m},\mathbf{V})}{\mathbb{P}(\mathbf{f}|\mathbf{y},\mathbf{X},\boldsymbol{\theta})}\mathrm{d}\mathbf{f}\\
&\overset{(4.4)}{=} \int\mathcal{N}(\mathbf{f}|\mathbf{m},\mathbf{V})\ln\frac{Z\cdot\mathcal{N}(\mathbf{f}|\mathbf{m},\mathbf{V})}{\mathcal{N}(\mathbf{f}|\mathbf{m},\mathbf{V})\prod_{i=1}^{n}\mathbb{P}(y_i|f_i)}\mathrm{d}\mathbf{f}\\
&= \ln Z + \int\mathcal{N}(\mathbf{f}|\mathbf{m},\mathbf{V})\ln\mathcal{N}(\mathbf{f}|\mathbf{m},\mathbf{V})\,\mathrm{d}\mathbf{f}\\
&\quad -\int\mathcal{N}(\mathbf{f}|\mathbf{m},\mathbf{V})\ln\prod_{i=1}^{n}\mathbb{P}(y_i|f_i)\mathrm{d}\mathbf{f}\\
&\quad -\int\mathcal{N}(\mathbf{f}|\mathbf{m},\mathbf{V})\ln\mathcal{N}(\mathbf{f}|\mathbf{0},\mathbf{K})\mathrm{d}\mathbf{f}.
\end{aligned}$$

There are three Gaussian integrals to evaluate; the entropy of the approximate posterior and two other expectations

$$\begin{aligned}
\mathrm{KL}\left(\mathbb{Q}(\mathbf{f}|\boldsymbol{\theta}) \;||\; \mathbb{P}(\mathbf{f}|\mathbf{y},\mathbf{X},\boldsymbol{\theta})\right) &= \ln Z - \frac{1}{2}\ln|\mathbf{V}| - \frac{n}{2} - \frac{n}{2}\ln 2\pi\\
&\quad -\int\mathcal{N}(f)\left[\sum_{i=1}^{n}\ln\mathrm{sig}\left(\sqrt{v_{ii}}y_i f + m_i y_i\right)\right]\mathrm{d}f \qquad \text{(F.3)}\\
&\quad +\frac{n}{2}\ln 2\pi + \frac{1}{2}\ln|\mathbf{K}| + \frac{1}{2}\mathbf{m}^\top\mathbf{K}^{-1}\mathbf{m} + \frac{1}{2}\mathrm{tr}\left(\mathbf{K}^{-1}\mathbf{V}\right).
\end{aligned}$$

Summing up and dropping the constant (w.r.t. **m** and **V**) terms, we arrive at

$$\mathrm{KL}(\mathbf{m},\mathbf{V}) \overset{c}{=} -\int\mathcal{N}(f)\left[\sum_{i=1}^{n}\ln\mathrm{sig}\left(\sqrt{v_{ii}}y_i f + m_i y_i\right)\right]\mathrm{d}f - \frac{1}{2}\ln|\mathbf{V}| + \frac{1}{2}\mathbf{m}^\top\mathbf{K}^{-1}\mathbf{m} + \frac{1}{2}\mathrm{tr}\left(\mathbf{K}^{-1}\mathbf{V}\right)$$

F.7 Gaussian integral for VB lower bound

$$\begin{aligned}
Z_{VB} &= \int \mathbb{P}\left(\mathbf{f}|\mathbf{X}\right) \mathbb{Q}\left(\mathbf{y}|\mathbf{f},\mathbf{A},\mathbf{b},\mathbf{c}\right) \mathrm{d}\mathbf{f} = \int \mathcal{N}(\mathbf{f}|0,\mathbf{K}) \exp\left(\mathbf{f}^\top\mathbf{A}\mathbf{f} + (\mathbf{b}\odot\mathbf{y})^\top \mathbf{f} + \mathbf{c}^\top\mathbf{1}\right) \mathrm{d}\mathbf{f} \\
&= \frac{\exp\left(\mathbf{c}^\top\mathbf{1}\right)}{\sqrt{(2\pi)^n\,|\mathbf{K}|}} \int \exp\left(-\frac{1}{2}\mathbf{f}^\top\left(\mathbf{K}^{-1} - 2\mathbf{A}\right)\mathbf{f} + (\mathbf{b}\odot\mathbf{y})^\top\mathbf{f}\right) \mathrm{d}\mathbf{f} \\
&= \frac{\exp\left(\mathbf{c}^\top\mathbf{1}\right)}{\sqrt{(2\pi)^n\,|\mathbf{K}|}} \sqrt{\frac{(2\pi)^n}{|\mathbf{K}^{-1} - 2\mathbf{A}|}} \exp\left(\frac{1}{2}(\mathbf{b}\odot\mathbf{y})^\top\left(\mathbf{K}^{-1} - 2\mathbf{A}\right)^{-1}(\mathbf{b}\odot\mathbf{y})\right) \\
&= \frac{\exp\left(\mathbf{c}^\top\mathbf{1}\right)}{\sqrt{|\mathbf{I} - 2\mathbf{A}\mathbf{K}|}} \exp\left(\frac{1}{2}(\mathbf{b}\odot\mathbf{y})^\top\left(\mathbf{K}^{-1} - 2\mathbf{A}\right)^{-1}(\mathbf{b}\odot\mathbf{y})\right) \\
\ln Z_{VB} &= \mathbf{c}^\top\mathbf{1} + \frac{1}{2}(\mathbf{b}\odot\mathbf{y})^\top\left(\mathbf{K}^{-1} - 2\mathbf{A}\right)^{-1}(\mathbf{b}\odot\mathbf{y}) - \frac{1}{2}\ln|\mathbf{I} - 2\mathbf{A}\mathbf{K}|
\end{aligned}$$

F.8 Lower bound for the cumulative Gaussian likelihood

A lower bound

$$\mathrm{sig}_{\mathrm{probit}}(y_i f_i) \quad \geq \quad \mathbb{Q}\left(y_i|f_i,\varsigma_i\right) \;=\; a_i f_i^2 + b_i f_i + c_i$$

for the cumulative Gaussian likelihood function is derived by matching the function at one point ς

$$\mathbb{Q}\left(y_i = +1|f_i,\varsigma_i\right) \quad = \quad \mathrm{sig}_{\mathrm{probit}}(\varsigma_i),\ \forall i$$

and by matching the first derivative

$$\left.\frac{\partial}{\partial f_i}\ln\mathbb{Q}\left(y_i = +1|f_i,\varsigma_i\right)\right|_{\varsigma_i} \quad = \quad \frac{\partial \ln \mathrm{sig}_{\mathrm{probit}}(y_i f_i)}{\partial f_i} = \frac{\mathcal{N}(\varsigma_i)}{\mathrm{sig}_{\mathrm{probit}}(\varsigma_i)},\ \forall i$$

at this point for a tight approximation. Solving for these constraints leads to the coefficients

$$\begin{aligned}
\text{asymptotic behavior} \Rightarrow a_i &= -\frac{1}{2} \\
\text{first derivative} \Rightarrow b_i &= \varsigma_i + \frac{\mathcal{N}(\varsigma_i)}{\mathrm{sig}_{\mathrm{probit}}(\varsigma_i)} \\
\text{point matching} \Rightarrow c_i &= \left(\frac{\varsigma_i}{2} - b_i\right)\varsigma_i + \log \mathrm{sig}_{\mathrm{probit}}(\varsigma_i).
\end{aligned}$$

F.9 Free form optimisation for FV

We make a factorial approximation $\mathbb{P}\left(\mathbf{f}|\mathbf{y},\mathbf{X}\right) \approx \mathbb{Q}\left(\mathbf{f}\right) := \prod_i \mathbb{Q}\left(f_i\right)$ to the posterior by minimising $\mathrm{KL}[\mathbb{Q}\left(\mathbf{f}\right)||\mathbb{P}\left(\mathbf{f}\right)]$.

$$\begin{aligned}
\mathrm{KL}[\mathbb{Q}\left(\mathbf{f}\right)||\mathbb{P}\left(\mathbf{f}\right)] &= \int \prod_{i=1}^n \mathbb{Q}\left(f_i\right) \ln \frac{Z \cdot \prod_{i=1}^n \mathbb{Q}\left(f_i\right)}{\mathcal{N}(\mathbf{f}|\mathbf{m},\mathbf{V}) \prod_{i=1}^n \mathbb{P}(y_i|f_i)} \mathrm{d}\mathbf{f} \\
&= \sum_i \int \mathbb{Q}\left(f_i\right) \ln \frac{\mathbb{Q}\left(f_i\right)}{\mathbb{P}\left(y_i|f_i\right)} \mathrm{d}f_i + \frac{1}{2}\int \prod_{i=1}^n \mathbb{Q}\left(f_i\right) \mathbf{f}^\top\mathbf{K}^{-1}\mathbf{f}\mathrm{d}\mathbf{f} + \mathrm{const}_{\mathbf{f}}
\end{aligned}$$

Free-form optimisation proceeds by equating the functional derivative with zero

$$\frac{\delta \mathrm{KL}}{\delta \mathbb{Q}(f_i)} = \ln \mathbb{Q}(f_i) + 1 - \ln \mathbb{P}(y_i|f_i) + \frac{1}{2}\frac{\delta}{\delta \mathbb{Q}(f_i)} \int \prod_{i=1}^{n} \mathbb{Q}(f_i)\, \mathbf{f}^\top \mathbf{K}^{-1}\mathbf{f}\mathrm{d}\mathbf{f}. \tag{F.4}$$

We abbreviate the integral in the last term with ξ and rewrite it in terms of simple one-dimensional integrals $m_l = \int f_l \mathbb{Q}(f_l)\,\mathrm{d}f_l$ and $v_l = \int f_l^2 \mathbb{Q}(f_l)\,\mathrm{d}f_l - m_l^2$

$$\begin{aligned}
\xi &= \int \prod_i \mathbb{Q}(f_i) \sum_{j,k} f_j \left[\mathbf{K}^{-1}\right]_{jk} f_k \mathrm{d}\mathbf{f} \\
&= \int \prod_{i\neq l} \mathbb{Q}(f_i) \left[\int \mathbb{Q}(f_l) \left(f_l^2 \left[\mathbf{K}^{-1}\right]_{ll} + 2 f_l \sum_{j\neq l} f_j \left[\mathbf{K}^{-1}\right]_{jl} + \sum_{j\neq l, k\neq l} f_j \left[\mathbf{K}^{-1}\right]_{jk} f_k \right) \mathrm{d}f_l \right] \mathrm{d}\mathbf{f}_{\neg l} \\
&= \int \prod_{i\neq l} \mathbb{Q}(f_i) \left[\left[\mathbf{K}^{-1}\right]_{ll} \underbrace{\int f_l^2 \mathbb{Q}(f_l)\,\mathrm{d}f_l}_{v_l + m_l^2} + 2 (\sum_{j\neq l} f_j \left[\mathbf{K}^{-1}\right]_{jl}) \underbrace{\int f_l \mathbb{Q}(f_l)\,\mathrm{d}f_l}_{m_l} + \sum_{j\neq l, k\neq l} f_j \left[\mathbf{K}^{-1}\right]_{jk} f_k \right] \mathrm{d}\mathbf{f}_{\neg l} \\
&= \left[\mathbf{K}^{-1}\right]_{ll} (v_l + m_l^2) + 2 \sum_{j\neq l} m_j \left[\mathbf{K}^{-1}\right]_{jl} m_l + \int \prod_{i\neq l} \mathbb{Q}(f_i) \sum_{j\neq l, k\neq l} f_j \left[\mathbf{K}^{-1}\right]_{jk} f_k \mathrm{d}\mathbf{f}_{\neg l} \\
&= \text{induction over } l \\
&= \sum_l \left[\mathbf{K}^{-1}\right]_{ll} (v_l + m_l^2) + 2 \sum_{j<l} m_j \left[\mathbf{K}^{-1}\right]_{jl} m_l.
\end{aligned}$$

Plugging this into equation F.4 and using $\frac{\delta \int f_l^p \mathbb{Q}(f_l)\mathrm{d}f_l}{\delta \mathbb{Q}(f_l)} = f_l^p$, we find

$$\begin{aligned}
\frac{\delta \mathrm{KL}}{\delta \mathbb{Q}(f_i)} &= \ln \mathbb{Q}(f_i) + 1 - \ln \mathbb{P}(y_i|f_i) + \frac{1}{2} f_i \left[\mathbf{K}^{-1}\right]_{ii} f_i + f_i \sum_l \left[\mathbf{K}^{-1}\right]_{il} m_l \overset{!}{\equiv} 0 \\
\Rightarrow \mathbb{Q}(f_i) &\propto \exp\left(-\frac{1}{2} f_i \left[\mathbf{K}^{-1}\right]_{ii} f_i - f_i \sum_{l\neq i} \left[\mathbf{K}^{-1}\right]_{il} m_l \right) \mathbb{P}(y_i|f_i) \\
\Rightarrow \mathbb{Q}(f_i) &\propto \mathcal{N}\left(f_i \middle| m_i - \frac{\left[\mathbf{K}^{-1}\mathbf{m}\right]_i}{\left[\mathbf{K}^{-1}\right]_{ii}}, \left[\mathbf{K}^{-1}\right]_{ii}^{-1} \right) \mathbb{P}(y_i|f_i)
\end{aligned}$$

as the functional form of the best possible factorial approximation, namely a product of the true likelihood times a Gaussian with the same precision as the prior marginal.

Appendix G

Adaptive Compressed Sensing of Natural Images

G.1 Failure of basis pursuit started from wavelet coefficients

In this section, we show that the reconstruction error $\epsilon = \|\hat{\mathbf{u}} - \mathbf{u}\|_2$ of the noise-free L_1 method (basis pursuit: $\hat{\mathbf{u}}_{BP} = \arg\min_{\mathbf{u}}\{\lambda\|\mathbf{u}\|_1 + \frac{1}{2}\|\mathbf{X}\mathbf{u} - \mathbf{y}\|_2^2\}$) without total variation (TV) term can increase with new measurements if we start from coarse scale wavelet measurements.

Since there is no TV term, we have $\mathbf{B} = \mathbf{W}$ with $\mathbf{W}^\top = \mathbf{W}^{-1}$ the wavelet transform matrix leading to $\hat{\mathbf{s}}_{BP} = \arg\min_{\mathbf{s}}\{\lambda\|\mathbf{s}\|_1 + \frac{1}{2}\|\mathbf{X}\mathbf{W}^\top\mathbf{s} - \mathbf{y}\|_2^2, \mathbf{s} = \mathbf{W}\mathbf{u}\}$. Initially, $\mathbf{X} = \mathbf{W}_I$, $\mathbf{y} = \mathbf{W}_I\mathbf{u} + \boldsymbol{\varepsilon}$, where I contains the coarse scale wavelet indices. Further, the corresponding initial estimate is $\hat{\mathbf{s}}_{BP} = \arg\min\{\lambda\|\mathbf{s}_I\|_1 + \lambda\|\mathbf{s}_{\neg I}\|_1 + \frac{1}{2}\|\mathbf{s}_I - \mathbf{y}\|_2^2\}$, thus $\hat{\mathbf{s}}_{\neg I} = \mathbf{0}$ and $\hat{\mathbf{s}}_I = \lambda \cdot \kappa(\lambda^{-1}\mathbf{y})$. Here, $\neg I$ is short for $\{1, .., n\} \setminus I$ and

$$\kappa(y) = \begin{cases} y - \operatorname{sign}(y) & |y| > 1 \\ 0 & |y| \leq 1 \end{cases} = \operatorname{sign}(y) \cdot \max\left(|y| - 1, 0\right)$$

is the *soft-thresholding rule* [Donoho and Johnstone, 1994]. For a new measurement along a unit norm vector $\mathbf{x}_*$, we define $\mathbf{v} = \mathbf{W}\mathbf{x}_*$ and $r = y_* - \mathbf{v}_I^\top\mathbf{s}_I = \mathbf{u}^\top\mathbf{x}_* - \mathbf{v}_I^\top\mathbf{s}_I$.

In the noise-free case of $\lambda \to 0$, the quadratic term dominates and hence $\hat{\mathbf{s}}_I = \mathbf{y}$ implying a squared error of $\epsilon^2 = \|\hat{\mathbf{s}} - \mathbf{W}\mathbf{u}\|_2^2 = \|\mathbf{W}_{\neg I}\mathbf{u}\|_2^2$. A new measurement $(\mathbf{x}_*, y_*)$ does not affect $\hat{\mathbf{s}}_I = \mathbf{y}$ and we have $\hat{\mathbf{s}}_{\neg I} = \arg\min\{\|\mathbf{s}_{\neg I}\|_1, \mathbf{v}_{\neg I}^\top\mathbf{s}_{\neg I} = r\}$ for the remaining coefficients. Note that the constraint $\mathbf{v}_{\neg I}^\top\mathbf{s}_{\neg I} = r$ can always be satisfied by rescaling $\mathbf{s}_{\neg I}$

$$\hat{\mathbf{s}}_{\neg I} \quad = \quad \arg\min_{\mathbf{s}_{\neg I}}\{\|\mathbf{s}_{\neg I}\|_1, \mathbf{v}_{\neg I}^\top\mathbf{s}_{\neg I} = r\} = \arg\min_{\mathbf{s}_{\neg I}}\{|r/(\mathbf{v}_{\neg I}^\top\mathbf{s}_{\neg I})| \cdot \|\mathbf{s}_{\neg I}\|_1\}.$$

To derive an expression for $\hat{\mathbf{s}}_{\neg I}$ and to simplify notation, we define $\tilde{\mathbf{s}} = \mathbf{s}_{\neg I}$ and $\tilde{\mathbf{v}} = \mathbf{v}_{\neg I}$. The minimum of $\|\tilde{\mathbf{s}}\|_1$, satisfying $\tilde{\mathbf{v}}^\top\tilde{\mathbf{s}} = r$, does exist. Assume that $\tilde{\mathbf{v}} \neq \mathbf{0}$ (otherwise $\tilde{\mathbf{s}} = \mathbf{0}$). Let $i = \arg\max|\tilde{v}_i|$ (then, $\tilde{v}_i \neq 0$). Suppose that $\tilde{s}_j \neq 0$ for $j \neq i$. Now,

$$\tilde{v}_i\tilde{s}_i + \tilde{v}_j\tilde{s}_j = \tilde{v}_i\left(\tilde{s}_i + \frac{\tilde{v}_j}{\tilde{v}_i}\tilde{s}_j\right) + \tilde{v}_j 0 \text{ and } \left|\tilde{s}_i + \frac{\tilde{v}_j}{\tilde{v}_i}\tilde{s}_j\right| \leq |\tilde{s}_i| + \left|\frac{\tilde{v}_j}{\tilde{v}_i}\right| |\tilde{s}_j| \leq |\tilde{s}_i| + |\tilde{s}_j|,$$

so that $\|\tilde{\mathbf{s}}\|_1$ is not increased by setting $\tilde{s}_j = 0$ that way. Therefore, the (unique if i is unique) minimiser is $\tilde{\mathbf{s}} = r/\tilde{v}_i\mathbf{e}_i = r\mathbf{e}_i \odot \tilde{\mathbf{v}}^{-1}$ or

$$\hat{\mathbf{s}}_{\neg I} = \frac{r}{[\mathbf{v}_{\neg I}]_i}\mathbf{e}_i,\ i = \arg\max_{j \in \neg I}\left|\mathbf{w}_j^\top\mathbf{x}_*\right|,$$

where $\mathbf{e}_i$ is the ith unit vector. The associated error

$$\begin{aligned}
\tilde{\epsilon}^2 \quad &= \quad \|\hat{\mathbf{s}} - \mathbf{W}\mathbf{u}\|_2^2 = \|\hat{\mathbf{s}}_{\neg I} - \mathbf{W}_{\neg I}\mathbf{u}\|_2^2 = \|\mathbf{W}_{\neg I}\mathbf{u}\|_2^2 + \hat{\mathbf{s}}_{\neg I}^\top\left(\hat{\mathbf{s}}_{\neg I} - 2\mathbf{W}_{\neg I}\mathbf{u}\right) \\
&= \quad \epsilon^2 + \frac{r}{[\mathbf{v}_{\neg I}]_i}\mathbf{e}_i^\top\left(\frac{r}{[\mathbf{v}_{\neg I}]_i}\mathbf{e}_i - 2\mathbf{W}_{\neg I}\mathbf{u}\right) \\
&= \quad \epsilon^2 + \frac{2r^2}{[\mathbf{v}_{\neg I}]_i^2}\left(\frac{1}{2} - \frac{[\mathbf{v}_{\neg I}]_i[\mathbf{W}_{\neg I}\mathbf{u}]_i}{r}\right)
\end{aligned}$$

does increase whenever $\mathbf{x}_*$ satisfies

$$\begin{aligned}
\tilde{\epsilon}^2 > \epsilon^2 \quad &\Leftrightarrow \quad \frac{[\mathbf{v}_{\neg I}]_i \cdot [\mathbf{W}_{\neg I}\mathbf{u}]_i}{r} = \frac{s_i \cdot v_i}{y_* - \mathbf{y}^\top \mathbf{W}_I \mathbf{x}_*} = \frac{s_i \cdot v_i}{\mathbf{u}^\top \mathbf{W}^\top \mathbf{W}\mathbf{x}_* - \mathbf{y}^\top \mathbf{v}_I} < \frac{1}{2} \\
&\Leftrightarrow \quad 2s_i \cdot v_i \mathrm{sign}(\mathbf{s}^\top \mathbf{v} - \mathbf{y}^\top \mathbf{v}_I) < |\mathbf{s}^\top \mathbf{v} - \mathbf{y}^\top \mathbf{v}_I| \\
&\Leftrightarrow \quad 2s_i v_i \cdot \mathrm{sign}(\mathbf{s}_{\neg I}^\top \mathbf{v}_{\neg I}) < |\mathbf{s}_{\neg I}^\top \mathbf{v}_{\neg I}|.
\end{aligned}$$

By choosing $\mathbf{v}_{\neg I} = \alpha^{-1}\mathbf{s}_{\neg I}^{-1}, \alpha = \|1/\mathbf{s}_{\neg I}\|_2 > 0$, we obtain $2 < n$ as a necessary condition to increase the reconstruction error.

Thus, by measuring along a specifically chosen projection $\mathbf{x}_*$, it is actually possible to increase the error even though we have a noise level of $\sigma = 0$. Hence, the reconstruction error $\epsilon = \|\hat{\mathbf{u}}_{BP} - \mathbf{u}\|_2$ of the basis pursuit estimator without total variation penalty $\hat{\mathbf{u}}_{BP}$ is not monotonic in the amount of information available about the unknown $\mathbf{u}$.

Abbreviations

ADATAP	Adaptive TAP, 23
ADF	Assumed Density Filtering, 23
BCS	Bayesian Compressive Sensing, 88
BFGS	Broyden-Fletcher-Goldfarb-Shanno, 37
CASL	Continuous Arterial Spin Labeling, 104
CCCP	Convex-ConCave programming, 38
CG	Conjugate Gradients, 15
CRB	Cramér-Rao Bound, 25
CS	Compressed Sensing and/or Compressive Sampling, 85
CV	Cross Validation, 6
DAG	Directed Acyclic Graph, 3
DC	Difference of Convex programming, 38
DFT	Discrete Fourier Transformation, 87
EC	Expectation Consistent, 23
EP	Expectation Propagation, 17
EPI	Echo Planar Imaging, 104
ERM	Empirical Risk Minimisation, 6
FFT	Fast Fourier Transformation, 103
FFTW	Fastest Fourier Transformation in the West, 117
FLASH	Fast Low Angle SHot, 104
FV	Factorial Variational approach, 17
GBP	Gaussian Belief Propagation, 15
GLM	Generalised Linear Model, 5
GP	Gaussian Process, 5
i.i.d.	independently and identically distributed, 86
IRLS	Iteratively Reweighted Least Squares, 11
JPEG	Joint Photographic Experts Group, 85

KL	Gaussian KL minimisation algorithm, 17
KL	Kullback-Leibler divergence, 16
LA	Laplace's Approximation, 17
LASSO	Least Absolute Shrinkage and Selection Operator, 9
LBP	Loopy Belief Propagation, 23
LCG	Linear Conjugate Gradients, 15
LR	Label Regression, 54
MAP	Maximum A Posteriori, 7
MCMC	Markov Chain Monte Carlo, 17
ML	Maximum Likelihood, 9
MRF	Markov Random Field, 51
MRI	Magnetic Resonance Imaging, 104
MVM	Matrix Vector Multiplication, 15
NFFT	Nonequispaced Fast Fourier Transformation, 106
OLS	Ordinary Least Squares, 9
PCA	Principal Component Analysis, 43
PLS	Penalised Least Squares, 9
PSF	Point Spread Function, 109
RARE	Rapid Acquisition with Relaxation Enhancement, 104
SBL	Sparse Bayesian Learning, 32
SLM	Sparse Linear Model, 45
SRM	Structural Risk Minimisation, 6
SVM	Support Vector Machine, 7
SWIFT	SWeep Imaging with Fourier Transformation, 104
TAP	Thouless, Anderson and Palmer method, 23
TE	Echo Time, 117
TSE	Turbo Spin Echo, 116
TV	Total Variation, 157
UTE	Ultrashort Echo Time (TE), 104
VB	Variational Bounds, 17
ZFDC	Zero Filling Density Compensation, 109

Index

Bibliography

Jonathan S. Abel. A bound on mean-square-estimate error. *IEEE Transactions on Information Theory*, 39(5):1675–1680, 1993. 25

Yasmin Altun, Thomas Hofmann, and Alexander J. Smola. Gaussian process classification for segmenting and annotating sequences. In *Proceedings of the 21st International Conference on Machine Learning*, 2004. 54

Anthony C. Atkinson and Alexander N. Donev. *Optimum Experimental Design*. Oxford University Press, 2002. 25

Hagai Attias. A variational Bayesian framework for graphical models. In *Advances in Neural Information Processing Systems 12*, 2000. 39

Edward W. Barankin. Locally best unbiased estimates. *Annals of Mathematical Statistics*, 20(4): 477–501, 1949. 25

Ole E. Barndorff-Nielsen and David R. Cox. *Inference and asymptotics*. Chapman & Hall/CRC, 1989. 138

M. Jésus Bayarri and Jim O. Berger. The interplay of Bayesian and frequentist analysis. *Statistical Science*, 19(1):58–80, 2004. 6

M. A. Bernstein, K. F. King, and X. J. Zhou. *Handbook of MRI Pulse Sequences*. Academic Press, 1st edition, 2004. 104, 106, 109, 116, 118, 122

A. Bhattacharyya. On some analogues of the amount of information and their use in statistical estimation. *Sankhya: The Indian Journal of Statistics*, 8:1–14, 1946. 25

Christopher M. Bishop. *Pattern Recognition and Machine Learning*. Springer, 2006. 7, 11

Kai T. Block, Martin Uecker, and Jens Frahm. Undersampled radial MRI with multiple coils: Iterative image reconstruction with a total variation constraint. *Magnetic Resonance in Medicine*, 57:1086–1098, 2007. 105, 111

Vladimir I. Bogachev. *Gaussian Measures*. American Mathematical Society, 1998. 139

George E. P. Box and George C. Tiao. *Bayesian Inference in Statistical Analysis*. John Wiley & Sons, 1973. 33

Stephen Boyd and Lieven Vandenberghe. *Convex Optimization*. Cambridge University Press, 2004. 22, 35, 36, 37, 123, 129, 139, 140

Leo Breiman. Better subset regression using the nonnegative garrote. *Technometrics*, 37:373–384, 1995. 9

Emmanuel Candès and Justin Romberg. Practical signal recovery from random projections. In *Proceedings of SPIE*, 2004. 89, 94

Emmanuel Candès, Justin Romberg, and Terence Tao. Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information. *IEEE Transactions on Information Theory*, 52:489–509, 2006. 85, 86, 88, 93, 96, 98, 100, 102, 105, 121, 122

François Caron and Arnaud Doucet. Sparse Bayesian nonparametric regression. In *Proceedings of the 25th International Conference on Machine Learning*, 2008. 33

Kathryn Chaloner and Isabella Verdinelli. Bayesian experimental design: A review. *Statistical Science*, 10(3):273–304, 1995. 25, 88, 122

David Chandler. *Introduction to modern statistical mechanics*. Oxford University Press, 1987. 20

Scott S. Chen, David L. Donoho, and Michael A. Saunders. Atomic decomposition by basis pursuit. *SIAM Journal on Scientific Computing*, 20(1):33–61, 1999. 93, 111

Wei Chu, Zoubin Ghahramani, Francesco Falciani, and David L. Wild. Biomarker discovery in microarray gene expression data with Gaussian processes. *Bioinformatics*, 21:3385–3393, 2005. 54

Ronald Coifman, F. Geshwind, and Yves Meyer. Noiselets. *Applied and Computational Harmonic Analysis*, 10:27–44, 2001. 88, 98

John B. Copas. Regression, prediction and shrinkage. *Journal of the Royal Statistical Society, Series B*, 45(2):311–354, 1983. 9

Thomas M. Cover and Joy A. Thomas. *Elements of Information Theory*. John Wiley & Sons, 2006. 26, 113

Harald Cramér. *Mathematical Methods of Statistics*. Princeton University Press, 1946. 25

Lehel Csató. *Gaussian Processes – Iterative Sparse Approximations*. PhD thesis, Aston University, 2002. 60

Lehel Csató and Manfred Opper. Sparse On-Line Gaussian Processes. *Neural Computation*, 2 (14):641–668, 2002. 58

Lehel Csató, Ernest Fokoué, Manfred Opper, and Bernhard Schottky. Efficient approaches to Gaussian process classification. In *Advances in Neural Information Processing Systems 12*, 2000. 53, 66, 84

Ingrid Daubechies. *Ten Lectures on Wavelets*. SIAM, 1992. 89

Mark A. Davenport, Richard G. Baraniuk, and Clayton D. Scott. Tuning support vector machines for minimax and neyman-pearson classification. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, to appear, 2010. 7

Shai Dekel. Adaptive compressed image sensing based on wavelet-trees. http://shaidekel.tripod.com/adaptiveCS.pdf, 2008. 88, 98, 100

Arthur P. Dempster, Nan Laird, and Donald B. Rubin. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society, Series B*, 39:1–38, 1977. 39

Peter J. Diggle, J. A. Tawn, and R. A. Moyeed. Model-based Geostatistics. *Applied Statistics*, 47 (3):299–350, 1998. 54

David L. Donoho. Compressed sensing. *IEEE Transactions on Information Theory*, 52(4):1289–1306, 2006a. 85, 86, 93, 96, 105, 121, 122

David L. Donoho. For most large underdetermined systems of linear equations, the minimal ell-1 norm solution is also the sparsest solution. *Communications on Pure and Applied Mathematics*, 59:797–829, 2006b. 93

David L. Donoho and Jain M. Johnstone. Ideal spatial adaptation by wavelet shrinkage. *Biometrika*, 81(3):425–455, 1994. 157

Simon Duane, A. D. Kennedy, Brian J. Pendleton, and Duncan Roweth. Hybrid monte carlo. *Physics Letters B*, 195(2):216–222, 1987. doi: http://dx.doi.org/10.1016/0370-2693(87)91197-X. 32

Marco F. Duarte, Mark Davenport, Dharmpal Takhar, Jason Laska, Ting Sun, Kevin Kelly, and Richard Baraniuk. Single pixel imaging via compressive sampling. *IEEE Signal Processing Magazine*, 25:83–91, March 2008. 96

Michael Elad. Optimized projections for compressed sensing. *IEEE Transactions on Signal Processing*, 55:5695–5702, 2007. 88, 102

Valerii V. Fedorov. *Theory of Optimal Experiments*. Academic Press, 1972. 122

Rob Fergus, Barun Singh, Aaron Hertzmann, Sam T. Roweis, and William T. Freeman. Removing camera shake from a single photograph. In *ACM Transactions on Graphics, SIGGRAPH 2006 Conference Proceedings*, volume 25, pages 787–794, 2006. 124

David J. Field. Relations between the statistics of natural images and the response properties of cortical cells. *Journal of the Optical Society of America*, 4:2379–2394, 1987. 98

Mário A. T. Figueiredo. Adaptive sparseness using jeffreys prior. In *Advances in Neural Information Processing Systems 14*, 2002. 33

Shai Fine and Katya Scheinberg. Efficient SVM training using low-rank kernel representations. *Journal of Machine Learning Research*, 2:243–264, 2001. 71

Ronald A. Fisher. *The Design of Experiments*. Macmillan, 1935. 25

Brendan J. Frey and David J. C. MacKay. A revolution: Belief propagation in graphs with cycles. In *Advances in Neural Information Processing Systems 10*, 1998. 23

Urs Gamper, Peter Boesiger, and Sebastian Kozerke. Compressed sensing in dynamic MRI. *Magnetic Resonance in Medicine*, 59:365–373, 2008. 105, 122

A. Garnaev and E. Gluskin. The widths of a euclidean ball. *Soviet Math. Dokl.*, 30:200–204, 1984. 85

Allen N. Garroway, Peter K. Grannell, and Peter Mansfield. Image formation in NMR by a selective irradiative pulse. *Journal of Physics C: Solid State Physics*, 7:L457–L462, 1974. 104

Sebastian Gerwinn, Jakob Macke, Matthias W. Seeger, and Matthias Bethge. Bayesian inference for spiking neuron models with a sparsity prior. In *Advances in Neural Information Processing Systems 20*, 2008. 88, 94

Mark N. Gibbs and David J. C. MacKay. Variational Gaussian Process Classifiers. *IEEE Transactions on Neural Networks*, 11(6):1458–1464, 2000. 22, 53, 54, 64, 65, 84

Mark Girolami. A variational method for learning sparse and overcomplete representations. *Neural Computation*, 13:2517–2532, 2001. 31, 34, 37, 38

Mark Girolami and Simon Rogers. Variational Bayesian Multinomial Probit Regression with Gaussian Process Priors. *Neural Computation*, 18:1790–1817, 2006. 66

Gene H. Golub and Charles F. van Loan. *Matrix Computations*. The Johns Hopkins University Press, London, 3rd edition, 1996. 15, 16, 43, 93

Peter J. Green. Iteratively reweighted least squares for maximum likelihood estimation, and some robust and resistant alternatives. *Journal of the Royal Statistical Society, Series B*, 46(2): 149–192, 1984. 11, 41

Andreas Greiser and Markus von Kienlin. Efficient k-space sampling by density-weighted phase-encoding. *Magnetic Resonance in Medicine*, 50(6):1266–75, 2003. 105

Mark A. Griswold, Peter M. Jakob, Robin M. Heidemann, Mathias Nittka, Vladimir Jellus, Jianmin Wang, Berthold Kiefer, and Axel Haase. Generalized autocalibrating partially parallel acquisitions (GRAPPA). *Magnetic Resonance in Medicine*, 47(6):1202–10, 2002. 104

A. Haase, J. Frahm, D. Matthaei, W. Hänicke, and K. Merboldt. FLASH imaging: Rapid NMR imaging using low flip-angle pulses. *Journal of Magnetic Resonance*, 67:258–266, 1986. 104

Trevor Hastie, Robert Tibshirani, and Jerome Friedman. *The Elements of Statistical Learning: Data Mining, Inference, and Prediction.* Springer, 2009. 9, 10

Lihan He and Lawrence Carin. Exploiting structure in wavelet-based Bayesian compressive sensing. *IEEE Transactions on Signal Processing*, 57(9):3488–3497, 2009. ISSN 1053-587X. doi: http://dx.doi.org/10.1109/TSP.2009.2022003. 88

Jürgen Hennig, A. Nauerth, and H. Friedburg. RARE imaging: A fast imaging method for clinical MR. *Magnetic Resonance in Medicine*, 3(6):823–833, 1986. 104

Magnus R. Hestenes and Eduard Stiefel. Methods of conjugate gradients for solving linear systems. *Journal of Research of the National Bureau of Standards*, 49(6):409–436, December 1952. 15

Simon Hu, Michael Lustig, Albert P. Chen, Jason Crane, Adam Kerr, Douglas A.C. Kelley, Ralph Hurd, John Kurhanewica, Sarah J. Nelson, John M. Pauly, and Daniel B. Vigneron. Compressed sensing for resolution enhancement of hyperpolarized 13C flyback 3D-MRSI. *Journal of Magnetic Resonance*, 192(2):258–264, 2008. 105

Marc M. Van Hulle. Edgeworth approximation of multivariate differential entropy. *Neural Computation*, 17:1903–1910, 2005. 138

Djaudat Idiyatullin, Curt Corum, Jang-Yeon Park, and Michael Garwood. Fast and quiet MRI using a swept radiofrequency. *Journal of Magnetic Resonance*, 181(2):342–349, 2006. 104

Hemant Ishwaran and J. Sunil Rao. Spike and slab gene selection for multigroup microarray data. *Journal of the American Statistical Association*, 100(471):764–780, 2005. 45

R. S. Ismagilov. Widths of sets in normed linear spaces and the approximation of functions by trigonometric polynomials. *Russian Math. Surveys*, 29:161–178, 1974. 85

Tomi S. Jaakkola. *Variational Methods for Inference and Estimation in Graphical Models.* PhD thesis, MIT, 1997. 31, 34, 37

Tomi S. Jaakkola and Michael I. Jordan. Computing upper and lower bounds on likelihoods in intractable networks. In *Uncertainty in Artificial Intelligence (UAI)*, 1996. 22, 53, 64, 84

Shihao Ji and Lawrence Carin. Bayesian compressive sensing and projection optimization. In *Proceedings of the 24th International Conference on Machine Learning*, 2007. 85, 88, 93, 94, 95, 96, 97, 100, 102

Michael I. Jordan. *Learning in Graphical Models.* Kluwer, 1997. 113

Michael I. Jordan. Are you a Bayesian or a frequentist? Summer School Lecture, Cambridge, 2009. URL `http://mlg.eng.cam.ac.uk/mlss09/mlss_slides/Jordan_1.pdf`. 1, 6

Michael I. Jordan, Zoubin Gharamani, Tomi S. Jaakkola, and Lawrence K. Saul. An introduction to variational methods for graphical models. *Machine Learning*, 37:183–233, 1999. 16

Ashish Kapoor and Rosalind W. Picard. Multimodal affect recognition in learning environments. In *ACM international conference on Multimedia*, 2005. 54

Ashish Kapoor, Kristen Grauman, Raquel Urtasun, and Trevor Darrell. Active learning with gaussian processes for object categorization. In *ICCV*, 2007. 54

Boris Sergeevich Kashin. Widths of certain finite-dimensional sets and classes of smooth functions. *Math. USSR IZV.*, 11:317–333, 1978. 85

Frank R. Korosec, Richard Frayne, Thomas M. Grist, and Charles A. Mistretta. Time-resolved contrast-enhanced 3D MR angiography. *Magnetic Resonance in Medicine*, 36:345–351, 1996. 105

Malte Kuss and Carl E. Rasmussen. Assessing Approximate Inference for Binary Gaussian Process Classification. *Journal of Machine Learning Research*, 6:1679 – 1704, 10 2005. 54, 76, 81, 82, 84

Christoph H. Lampert, Hannes Nickisch, and Stefan Harmeling. Learning to detect unseen object classes by between-class attribute transfer. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2009)*, 2009. 4

Cornelius Lanczos. An iteration method for the solution of the eigenvalue problem of linear differential and integral operators. *Journal of Research of the National Bureau of Standards*, 45 (4):255–282, October 1950. 16, 41, 93

Steffen L. Lauritzen. *Graphical Models*. Oxford Statistical Sciences. Oxford Statistical Science Series, 1996. 12

Paul C. Lauterbur. Image formation by induced local interactions: Examples employing nuclear magnetic resonance. *Nature*, 242:190–191, 1973. 104

Neil D. Lawrence, Matthias W. Seeger, and Ralf Herbrich. Fast sparse Gaussian process methods: The informative vector machine. In *Advances in Neural Information Processing Systems 16*, 2004. 60

Erich L. Lehmann and George Casella. *Theory of Point Estimation*. Texts in Statistics. Springer Texts in Statistics, 1st edition, 1998. 89, 92

Adrian S. Lewis. Derivatives of spectral functions. *Mathematics of Operations Research*, 21:576–588, 1996. 127

Michael Lustig, David L. Donoho, and John M. Pauly. Sparse MRI: The application of compressed sensing for rapid MR imaging. *Magnetic Resonance in Medicine*, 85(6):1182–1195, 2007. 103, 104, 111, 112, 116, 117, 119, 121

Helmut Lütkepohl. *Handbook of Matrices*. John Wiley & Sons, December 1997. 125, 126

David J. C. MacKay. Bayesian Interpolation. *Neural Computation*, 4(3):415–447, 1992. 60

David J. C. MacKay. *Information Theory, Inference, and Learning Algorithms*. Cambridge University Press, Cambridge, UK, 3rd edition, September 2005. 7

Bruno Madore, Gary H. Glover, and Norbert J. Pelc. Unalising by Fourier-encoding the overlaps using the temporal dimension (UNFOLD), applied to cardiac imaging and fMRI. *Magnetic Resonance in Medicine*, 42:813–828, 1999. 105

Jan R. Magnus and Heinz Neudecker. *Matrix Differential Calculus*. John Wiley & Sons, 1999. 126

Dmitry M. Malioutov, Jason K. Johnson, and Alan S. Willsky. Low-rank variance estimation in large-scale GMRF models. In *International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2006a. 51

Dmitry M. Malioutov, Jason K. Johnson, and Alan S. Willsky. Walk-sums and belief propagation in Gaussian graphical models. *Journal of Machine Learning Research*, 7:2031–2064, 2006b. 51

Peter Mansfield. Multi-planar image formation using NMR spin-echoes. *Journal of Physics C: Solid State Physics*, 10:L50–L58, 1977. 104

G. Marseille, R. de Beer, M. Fuderer, A. Mehlkopf, and D. van Ormondt. Nonuniform phase-encode distributions for MRI scan time reduction. *Journal of Magnetic Resonance*, 111(1):70–75, 1996. 104, 105, 122

Georges F. Matheron. The intrinsic random functions and their applications. *Advances in Applied Probability*, 5:439–468, 1973. 11

Peter S. Maybeck. *Stochastic Models, Estimation and Control*. Academic Press, 1982. 23

Peter McCullagh. *Tensor Methods in Statistics*. Chapman & Hall/CRC, 1987. 138

Peter McCullagh and John Nelder. *Generalized Linear Models*. London: Chapman and Hall, 1989. 10

G. McGibney, M. R. Smith, S. T. Nichols, and A. Crawley. Quantitative evaluation of several partial Fourier reconstruction algorithms used in MRI. *Magnetic Resonance in Medicine*, 30(1): 51–9, 1993. 104

Thomas P. Minka. Expectation Propagation for Approximate Bayesian Inference. In *UAI*, pages 362–369. Morgan Kaufmann, 2001a. 23, 53, 62, 63, 88, 90

Thomas P. Minka. *A Family of Algorithms for Approximate Bayesian Inference*. PhD thesis, Department of Electrical Engineering and Computer Science, MIT, 2001b. 62

Thomas P. Minka. Divergence measures and message passing. Technical report, Microsoft Research, 2005. 16, 24, 68

Thomas P. Minka, John M. Winn, John P. Guiver, and Anitha Kannan. Infer.NET 2.3, 2009. Microsoft Research Cambridge. http://research.microsoft.com/infernet. 14

James W. Miskin. *Ensemble Learning for Independent Component Analysis*. PhD thesis, University of Cambridge, 2000. 15, 20, 66, 124

Charles A. Mistretta, O. Wieben, J. Velikina, W. Block, J. Perry, Y. Wu, K. Johnson, and Y. Wu. Highly constrained backprojection for time-resolved MRI. *Magnetic Resonance in Medicine*, 55:30–40, 2006. 105

Kevin Murphy, Yair Weiss, and Michael Jordan. Loopy belief propagation for approximate inference: An empirical study. In *Uncertainty in Artificial Intelligence (UAI)*, 1999. 23

Radford M. Neal. Probabilistic Inference Using Markov Chain Monte Carlo Methods. Technical Report CRG-TR-93-1, Department of Computer Science, University of Toronto, September 1993. 32, 68, 69

Radford M. Neal. Annealed Importance Sampling. *Statistics and Computing*, 11:125–139, 2001. 68, 69

John Nelder and Robert Wedderburn. Generalized linear models. *Journal of the Royal Statistical Society, Series A*, 135(3):370–384, 1972. 10

Hannes Nickisch and Carl E. Rasmussen. Approximations for binary Gaussian process classification. *Journal of Machine Learning Research*, 9:2035–2078, 10 2008. 4, 22, 53

Hannes Nickisch and Carl E. Rasmussen. Gaussian mixture density modeling with gplvms. In *32nd Annual Symposium of the German Association for Pattern Recognition (DAGM)*, in press. URL `http://mloss.org/software/view/272/`. 4

Hannes Nickisch and Matthias Seeger. Convex variational Bayesian inference for large scale generalized linear models. In *Proceedings of the 26th International Conference on Machine Learning*, 2009. 4, 31, 65

Hannes Nickisch, Pushmeet Kohli, and Carsten Rother. Learning an interactive segmentation system. In *Indian Conference on Computer Vision, Graphics and Image Processing (ICVGIP)*, accepted. URL `http://arxiv.org/abs/0912.2492`. 4

Manfred Opper and Cédric Archambeau. The Variational Gaussian Approximation Revisited. *Neural Computation*, 21(3):786–792, March 2009. 21, 22, 53, 63, 68

Manfred Opper and Ole Winther. Gaussian Processes for Classification: Mean Field Algorithms. *Neural Computation*, 12(11):2655–2684, 2000. 23, 63

Manfred Opper and Ole Winther. Expectation Consistent Approximate Inference. *Journal of Machine Learning Research*, 6:2177–2204, 2005. 17, 23, 24, 62, 68

Christopher C. Paige. Error analysis of the lanczos algorithm for tridiagonalizing a symmetric matrix. *Journal of Applied Mathematics*, 18(3):341–349, 1976. 43

Christopher C. Paige and Michael A. Saunders. LSQR: An algorithm for sparse linear equations and sparse least squares. *ACM Transactions on Mathematical Software*, 8(1):43–71, 1982. 109

Jason A. Palmer, David P. Wipf, Ken Kreutz-Delgado, and Bhaskar D. Rao. Variational EM Algorithms for non-Gaussian latent variable Models. In *Advances in Neural Information Processing Systems 18*, 2006. 18, 22, 31, 32, 34, 35, 37, 141

Giorgio Parisi. *Statistical field theory*. Addison-Wesley, 1988. 20

Beresford N. Parlett and D. S. Scott. The lanczos algorithm with selective orthogonalization. *Mathematics of Computation*, 33(145):217–238, 1979. 43

William H. Press, Saul A. Teukolsky, William T. Vetterling, and Brian P. Flannery. *Numerical Recipes in C*. Cambridge University Press, 2nd edition, February 1993. 61

Klaas P. Pruessmann, Markus Weiger, Markus B. Scheidegger, and Peter Boesiger. SENSE: Sensitivity encoding for fast MRI. *Magnetic Resonance in Medicine*, 42:952–962, 1999. 104

Friedrich Pukelsheim. *Optimal Design of Experiments*. SIAM Classics in Applied Mathematics 50, 2006. 25

Calyampudi R. Rao. Information and the accuracy attainable in the estimation of statistical parameters. *Bulletin of the Calcutta Mathematical Society*, 37:81–89, 1945. 25

Carl E. Rasmussen and Hannes Nickisch. Gaussian processes for machine learning toolbox. *Journal of Machine Learning Research*, accepted, August 2010. 4, 53, 70

Carl E. Rasmussen and Joaquin Quiñonero-Candela. Healing the Relevance Vector Machine through Augmentation. In *Proceedings of the 22nd International Conference on Machine Learning*, 2005. 60

Carl E. Rasmussen and Christopher K. I. Williams. *Gaussian Processes for Machine Learning*. The MIT Press, Cambridge, MA, 2006. 12, 24, 55, 56, 59, 61, 62, 67, 70

Ryan Rifkin and Aldebaro Klautau. In defense of one-vs-all classification. *Journal of Machine Learning Research*, 5:101–141, 2004. 66

Brian D. Ripley. *Spatial Statistics*. John Wiley & Sons, 1981. 11

Matthew D. Robson, Peter D. Gatehouse, Mark Bydder, and Graeme Bydder. Magnetic resonance: An introduction to ultrashort TE (UTE) imaging. *Journal of Computer Assisted Tomography*, 27(6):825–846, 2003. 104

Ralph T. Rockafellar. *Convex Analysis*. Princeton University Press, 1970. 34, 129, 141

Juan M. Santos, Charles H. Cunningham, Michael Lustig, Brian A. Hargreaves, Bob S. Hu, Dwight G. Nishimura, and John M. Pauly. Single breath-hold whole-heart MRA using variable-density spirals at 3T. *Magnetic Resonance in Medicine*, 55:371–379, 2006. 105

Andrew I. Schein and Lyle H. Ungar. Active learning for logistic regression: An evaluation. *Machine Learning*, 68:235–265, 2007. 49

Mark J. Schervish. *Theory of Statistics*. Springer, 1995. 6

Michael K. Schneider and Alan S. Willsky. Krylov subspace estimation. *SIAM Journal on Scientific Computing*, 22(5):1840–1864, 2001. 16, 41

Anton Schwaighofer, Volker Tresp, Peter Mayer, Alexander K. Scheel, and Gerhard Müller. The RA scanner: Prediction of rheumatoid joint inflammation based on laser imaging. In *Advances in Neural Information Processing Systems 15*, 2003. 54

Bernhard Schölkopf and Alexander J. Smola. *Learning with Kernels*. The MIT Press, Cambridge, MA, 2002. 7, 12

Matthias W. Seeger. Bayesian methods for support vector machines and Gaussian processes. Master's thesis, Universität Karlsruhe, 1999. 64

Matthias W. Seeger. *Bayesian Gaussian Process Models: PAC-Bayesian Generalisation Error Bounds and Sparse Approximations*. PhD thesis, University of Edinburgh, 2003. 21, 60, 133

Matthias W. Seeger. Gaussian processes for machine learning. *International Journal of Neural Systems*, 14(2):69–106, 2004. 12

Matthias W. Seeger. Bayesian inference and optimal design for the sparse linear mode. *Journal of Machine Learning Research*, 9:759–813, 2008. 24, 89, 92

Matthias W. Seeger. Gaussian covariance and scalable variational inference. In *Proceedings of the 27th International Conference on Machine Learning*, 2010a. 43

Matthias W. Seeger. Speeding up magnetic resonance image acquisition by Bayesian multi-slice adaptive compressed sensing. In *Advances in Neural Information Processing Systems 22*, pages 1633–1641, 2010b. 103, 122

Matthias W. Seeger and Hannes Nickisch. Compressed sensing and Bayesian experimental design. In *Proceedings of the 25th International Conference on Machine Learning*, 2008a. 4, 85

Matthias W. Seeger and Hannes Nickisch. Large scale variational inference and experimental design for sparse generalized linear models. Technical Report 175, Max Planck Institute for Biological Cybernetics, 9 2008b. 4, 31

Matthias W. Seeger and Hannes Nickisch. Large scale bayesian inference and experimental design for sparse linear models. Technical report, arXiv, 2010. URL `http://arxiv.org/abs/0810.0901`. 4

Matthias W. Seeger and Hannes Nickisch. Large scale Bayesian inference and experimental design for sparse linear models. *SIAM Journal on Imaging Sciences*, submitted. 4

Matthias W. Seeger, Florian Steinke, and Koji Tsuda. Bayesian inference and optimal design in the sparse linear model. In *International Conference on Artificial Intelligence and Statistics*, 2007. 95

Matthias W. Seeger, Hannes Nickisch, Rolf Pohmann, and Bernhard Schölkopf. Bayesian experimental design of magnetic resonance imaging sequences. In *Advances in Neural Information Processing Systems 21*, pages 1441–1448, 2009. 4, 104

Matthias W. Seeger, Hannes Nickisch, Rolf Pohmann, and Bernhard Schölkopf. Optimization of *k*-space trajectories for compressed sensing by Bayesian experimental design. *Magnetic Resonance in Medicine*, 63(1):116–126, 2010. doi: 10.1002/mrm.22180. 4, 104

Claude E. Shannon. A mathematical theory of communication. *Bell System Technical Journal*, 27: 379–423 and 623–656, July and October 1948. 86

Eero P. Simoncelli. Modeling the joint statistics of images in the Wavelet domain. In *Proceedings 44th SPIE*, pages 188–195, 1999. 87, 88, 89, 110

Alexander J. Smola and Bernhard Schölkopf. Sparse greedy matrix approximation for machine learning. In *Proceedings of the 17th International Conference on Machine Learning*, 2000. 71

Edward L. Snelson, Carl E. Rasmussen, and Zoubin Ghahramani. Warped Gaussian processes. In *Advances in Neural Information Processing Systems 16*, 2004. 54

Daniel K. Sodickson and Warren J. Manning. Simultaneous acquisition of spatial harmonics (SMASH): Fast imaging with radiofrequency coil arrays. *Magnetic Resonance in Medicine*, 38 (4):591–603, 1997. 104

Peter Sollich. Bayesian methods for support vector machines. *Machine Learning*, 46:21–52, 2002. 60

Daniel M. Spielman, John M. Pauly, and Craig H. Meyer. Magnetic resonance fluoroscopy using spirals with variable sampling densities. *Magnetic Resonance in Medicine*, 34(3):388–94, 1995. 105

Leonard A. Stefanski. A normal scale mixture representation of the logistic distribution. *Statistics & Probability Letters*, 11:69–70, 1990. 33

Charles Stein. Inadmissibility of the usual estimator for the mean of a multivariate distribution. In *Proceedings of the Third Berkeley Symposium on Mathematical Statistics and Probability*, pages 197–206, 1956. 9

S. Sundararajan and S. Sathiya Keerthi. Predictive approaches for choosing hyperparameters in Gaussian processes. *Neural Computation*, 13:1103–1118, 2001. 66

Robert J. Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society, Series B*, 58:267–288, 1996. 9, 94, 110, 111

Andrey N. Tikhonov and V. Y. Arsenin. *Solutions of ill posed problems*. John Wiley & Sons, 1977. 9

Michael E. Tipping. Sparse Bayesian Learning and the Relevance Vector Machine. *Journal of Machine Learning Research*, 1:211–244, 2001. 32, 33, 46, 60, 88, 89, 94, 102

Jeffrey Tsao, Peter Boesiger, and Klaas P. Pruessmann. k-t BLAST and k-t SENSE: Dynamic MRI with high frame rate exploting spatiotemporal correlations. *Magnetic Resonance in Medicine*, 50:1031–1042, 2003. 105

Vladimir N. Vapnik. *Statistical Learning Theory*. John Wiley & Sons, New York, 1998. 6

Markus von Kienlin and Raymond Mejia. Spectral localization with optimal point spread function. *Journal of Magnetic Resonance*, 94(2):268–287, 1991. 105

Martin J. Wainwright and Michael I. Jordan. Graphical models, exponential families, and variational inference. *Foundations and Trends in Machine Learning*, 1(1–2):1–305, 2008. doi: http:/dx.doi.org/10.1561/2200000001. 37, 132, 133

Martin J. Wainwright and Eero P. Simoncelli. Scale mixtures of gaussians and the statistics of natural images. In *Advances in Neural Information Processing Systems 12*, 2000. 33

Frank Wajer. *Non-Cartesian MRI Scan Time Reduction through Sparse Sampling*. PhD thesis, Delft University of Technology, 2001. 104, 105

Zhongmin Wang, Gonzalo R. Arce, and Jose L. Paredes. Colored projections for compressed sensing. In *ICASSP*, 2007. 88, 98

Larry Wasserman. *All of Statistics*. Springer, 2005. 6, 132

John B. Weaver, Yansun Xu, Dennis M. Healy Jr., and L. D. Cromwell. Filtering noise from images with wavelet transforms. *Magnetic Resonance in Medicine*, 21(2):288–295, 1991. 104, 105

Yair Weiss, , Hyun S. Chang, and William T. Freeman. Learning compressed sensing. Snowbird Learning Workshop, Allerton, CA, 2007. 85, 86, 96, 101

Mike West. On scale mixtures of normal distributions. *Biometrika*, 74(3):646–648, September 1987. 33

Christopher K. I. Williams and D. Barber. Bayesian Classification with Gaussian Processes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 12(20):1342–1351, 1998. 53, 56, 61

Christopher K. I. Williams and Carl Edward Rasmussen. Gaussian processes for regression. In *Advances in Neural Information Processing Systems 8*, 1996. 11

Christopher K. I. Williams and Matthias W. Seeger. Using the nyström method to speed up kernel machines. In *Advances in Neural Information Processing Systems 13*, 2001. 71

David S. Williams, John A. Detre, John S. Leigh, and Alan P. Koretsky. Magnetic resonance imaging of perfusion using spin inversion of arterial water. *The Proceedings of the National Academy of Sciences Online (US)*, 89:212–216, 1992. 104

John M. Winn and Christopher M. Bishop. Variational message passing. *Journal of Machine Learning Research*, 6:661–694, 2005. 24

David P. Wipf and Srikantan S. Nagarajan. A new view of automatic relevance determination. In *Advances in Neural Information Processing Systems 20*, 2008. 46

David P. Wipf, Bhaskar D. Rao, and Srikantan S. Nagarajan. Latent variable Bayesian models for promoting sparsity. 2010. URL `http://dsp.ucsd.edu/~dwipf/wipf_draft2009.pdf`. 124

David H. Wolpert. The lack of a priori distinctions between learning algorithms. *Neural Computation*, 8(7):1341–1390, 1996. 1

Max A. Woodbury. Inverting modified matrices. Memorandum 42, Statistical Research Group, Princeton University, Princeton, 1950. 125

Jong C. Ye, Sungho Tak, Yeji Han, and Hyun W. Park. Projection reconstruction MR imaging using FOCUSS. *Magnetic Resonance in Medicine*, 57:764–775, 2007. 105

Kai Yu, Jinbo Bi, and Volker Tresp. Active learning via transductive experimental design. In *Proceedings of the 23rd International Conference on Machine Learning*, 2006. 27

Kai Yu, Shenghuo Zhu, Wei Xu, and Yihong Gong. Non-greedy active learning for text categorization using convex ansductive experimental design. In *Proceedings of the 31st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 635–643, 2008. 27

Alan L. Yuille and Anand Rangarajan. The concave-convex procedure. *Neural Computation*, 15 (4):915–936, 2003. 39

Mingjun Zhong, Fabien Lotte, Mark Girolami, and Anatole Lécuyer. Classifying EEG for brain computer interfaces using Gaussian processes. *Pattern Recognition Letters*, 29:354–359, 2008. 54