Contents

Appendices

List of Figures

columnwidth 421.10046pt

1 Introduction

1.1 Motivation

- ozone coverage
- regularisation approach in atmospheric physics
- hierarchical modelling

1.2 What is going on?, 3 facts, What is new in this thesis?

- physical based hierarchical Bayesian model, sampling to TT approx
- RTE as an example
- non-linear to linear affine approximation

1.3 Thesis Outline

2
Theoretical and Technical Background

In this chapter, we provide a brief introduction to the methods used in this thesis. We keep it as general as possible, as more specific details will be presented in the results Chapter 3. We begin by introducing the forward model in Section 2.1, which we use to simulate the data. Since the forward model is weakly non-linear, we employ an affine transformation, see Section 2.2, to project the linear model onto the non-linear one, allowing us to treat the problem as a linear inverse problem. This enables the application of Bayesian inference in Section 2.3, where we formulate a hierarchical linear-Gaussian model to define and structure the posterior distribution. For comparison, we briefly present the Tikhonov regularization approach, see Section 2.4. In Section 2.5, we introduce Markov Chain Monte Carlo (MCMC) methods to sample from the posterior distribution. Finally, in Section 2.6, instead of sampling, we can approximate the posterior distribution using the tensor train (TT) format.

2.1 Forward Model

The forward model is based on a satellite measuring thermal radiation along its line of sight by pointing through the atmosphere, known as limb sounding, as shown in Figure 2.1. One measurement y_j, of a stationary satellite is the path integral through the atmosphere along the line of sight. For each measurement $j = 1, 2, \dots, m$ of a data set, we can define a tangent height h_{ℓ_j} as the shortest distance along the line of sight to the earth.

The j^{th} measurement, is modelled by the radiative transfer equation (RTE) [1]

$$y_j = \int_{\Gamma_j} B(\nu, T) k(\nu, T) \frac{p(T)}{k_{\text{B}} T(r)} x(r) \tau(r) \text{d}r + \eta_j \tag{2.1}$$

$$\tau(r) = \exp\left\{ - \int_{r_{\text{obs}}}^{r} k(\nu, T) \frac{p(T)}{k_B T(r')} x(r') \text{d}r' \right\}, \tag{2.2}$$

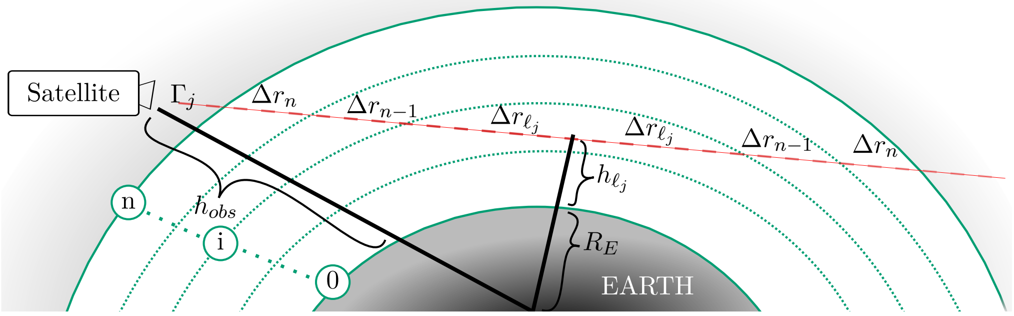


Figure 2.1: This figure illustrates a limb-sounding measurement setup, specifically how the line of sight of a satellite at altitude h_{obs} is partitioned according to a discretised atmospheric model. The atmosphere is divided into n layers, allowing the line of sight Γ_j to be discretised into segments Δr_i for $i = \ell_j, \dots, n$. Here, $\ell_j \in \mathbb{N}$ denotes the index corresponding to the tangent height h_{ℓ_j} relative to the Earth's radius R_E. This setup forms the basis for the numerical solution of the integral in Eq. 2.1, known as the radiative transfer equation.

where the path from the satellite along the line-of-sight of the j^{th} pointing direction is Γ_j and the ozone concentration at distance r from the radiometer is $x(r)$ plus some noise η_j. Within the atmosphere the number density $p(T)/(k_{\text{B}}T(r))$ of molecules is dependent on the pressure $p(T)$, the temperature $T(r)$, and the Boltzmann constant k_{B}. The factor $\tau(r) \leq 1$ accounts for re-absorption of the radiation along the line-of-sight, which makes the RTE non linear. The absorption constant $k(\nu, T)$ for a single gas molecule at a specific wavenumber ν is calculated according to the HITRAN database [2] and acts as a source function when multiplied with the black body radiation $B(\nu, T)$, given by Planck‘s law [3]. For fundamentals on the Radiative transfer equation we recommend [3, Chapter 1].

To enable matrix-vector multiplication, we discretise the atmosphere in n layers, where the i^{th} layer is defined by two spheres of radii $h_{i-1} < h_i$, for $i = 1, \dots, n$, with h_0 and h_n. Then we can discretise the ozone, pressure and temperature profiles as a function of height, where in between the heights h_{i-1} and h_i, each of the ozone concentration x_i, the pressure p_i, the temperature T_i, as well as the thermal radiation is assumed to be constant. Above h_n and below h_0, the ozone concentration is set to zero, so no signal can be obtained. Depending on the parameter of interest, which is either the ozone volume mixing ratio $\boldsymbol{x} = \{x_1, x_2, \dots, x_n\} \in \mathbb{R}^n$ or the fraction of pressure and temperature $\boldsymbol{p/T} = \{p_1/T_1, p_2/T_2, \dots, p_n/T_n\} \in \mathbb{R}^n$, we can rewrite the integral in Eq. (2.1) as e.g. as a vector multiplication $\boldsymbol{A_j}(\boldsymbol{x}, \boldsymbol{p}, \boldsymbol{T})\,\boldsymbol{x}$ or $\boldsymbol{A_j}(\boldsymbol{x}, \boldsymbol{p}, \boldsymbol{T})\,\boldsymbol{p/T}$, where the non-linear absorption $\tau(r)$ is included in $\boldsymbol{A_j}(\boldsymbol{x}, \boldsymbol{p}, \boldsymbol{T})$. Here, the row vector $\boldsymbol{A_j}(\boldsymbol{x}, \boldsymbol{p}, \boldsymbol{T}) \in \mathbb{R}^n$ defines a Kernel for each measurement so that the data vector

$$\boldsymbol{y} = \boldsymbol{A}(\boldsymbol{x}, \boldsymbol{p}, \boldsymbol{T})\,\boldsymbol{x} + \boldsymbol{\eta} = \boldsymbol{A}(\boldsymbol{x}, \boldsymbol{p}, \boldsymbol{T})\,\frac{\boldsymbol{p}}{\boldsymbol{T}} + \boldsymbol{\eta}\,. \tag{2.3}$$

can be written as a matrix-vector multiplication, with the matrix $\boldsymbol{A}(\boldsymbol{x},\boldsymbol{p},\boldsymbol{T}) \in \mathbb{R}^{m\times n}$ and the noise vector $\boldsymbol{\eta} \in \mathbb{R}^{m}$. Note, for simplicity, we do not explicitly specify whether $\boldsymbol{A}(\boldsymbol{x},\boldsymbol{p},\boldsymbol{T})$ is constructed under the assumption that $\boldsymbol{x}$ or $\boldsymbol{p}/\boldsymbol{T}$ is the parameter of interest..

Since the measurement process includes absorption $\tau(r)$, which reduces measurements only slightly, we classify the inverse problem as weakly non-linear. Hence, we can approximate the non-linear forward model $\boldsymbol{A}(\boldsymbol{x},\boldsymbol{p},\boldsymbol{T})$ with a map $\boldsymbol{M}$ and the linear forward model $\boldsymbol{A}_L$, so that $\boldsymbol{A}(\boldsymbol{x},\boldsymbol{p},\boldsymbol{T}) \approx \boldsymbol{M}\boldsymbol{A}_L$. Here, $\boldsymbol{A}_{L,j}$ of matrix $\boldsymbol{A}_L \in \mathbb{R}^{m\times n}$ is defined by the linear forward model, where absorption is neglected, e.g. set $\tau = 1$ in Eq. (2.2). Then each entry in the row vector $\boldsymbol{A}_{L,j}$ is either defined by $B(\nu,T)S(\nu,T)\frac{\boldsymbol{p}(T)}{k_{\mathrm{B}}\boldsymbol{T}(r)}\mathrm{d}r$ or $B(\nu,T)S(\nu,T)\frac{\boldsymbol{x}}{k_{\mathrm{B}}}\mathrm{d}r$, as in Eq. (2.1), depending on which parameter we focus on. This poses a linear inverse problem with the forward map defined by the matrix $\boldsymbol{A} = \boldsymbol{M}\boldsymbol{A}_L$, where $\boldsymbol{M}$ is, more specifically, an affine map.

2.2 Affine Map

An affine map is any linear map between two vector spaces or affine spaces, where an affine space does not need to preserve a zero origin, see [4, Def. 2.3.1]. In other words, an affine map does not need to map to the origin of the associated vector space or is a linear map on vector spaces including a translation, or in the words of my supervisor, C. F., an affine map is a Taylor series of first order. For more information on affine spaces and maps, we refer to the books [4, 5]

linear forward model
V
affine Map
M
W
non-linear forward model

$$\boldsymbol{A}(\boldsymbol{x},\boldsymbol{p},\boldsymbol{T})\boldsymbol{x} \approx \boldsymbol{M}\boldsymbol{A}_L\boldsymbol{x} = \boldsymbol{A}\boldsymbol{x}$$

Figure 2.2: This Figure shows the schematic representation of the affine map $\boldsymbol{M}$, which approximates the non-linear forward model from the linear forward model. Here, V contains values produced by the linear forward model, and W contains the corresponding values from the non-linear forward model. Both V and W are affine subspaces over the same field. The affine map $\boldsymbol{M}$ projects elements from the linear forward model space V onto their counterparts in the non-linear forward model space W.

Consequently, we introduce an affine map $\boldsymbol{M} : \boldsymbol{A}_L\boldsymbol{x} \to \boldsymbol{A}(\boldsymbol{x},\boldsymbol{p},\boldsymbol{T})\boldsymbol{x}$, which maps the linear forward model $\boldsymbol{A}_L\boldsymbol{x}$ onto the non-linear forward model $\boldsymbol{A}(\boldsymbol{x},\boldsymbol{p},\boldsymbol{T})\boldsymbol{x}$. Then the non-linear forward model matrix is approximated by $\boldsymbol{A}(\boldsymbol{x},\boldsymbol{p},\boldsymbol{T}) \approx \boldsymbol{M}\boldsymbol{A}_L$. In practise we generate two affine subspaces spaces $V = \{\boldsymbol{A}(\boldsymbol{x}^{(1)},\boldsymbol{p},\boldsymbol{T}),\dots,\boldsymbol{A}(\boldsymbol{x}^{(m)},\boldsymbol{p},\boldsymbol{T})\}$ and $W = \{\boldsymbol{A}_L\boldsymbol{x}^{(1)},\dots,\boldsymbol{A}_L\boldsymbol{x}^{(m)}\}$ over the same field, with fixed $\boldsymbol{p},\boldsymbol{T}$ and find the mapping in between those. Here, the parameter $\boldsymbol{x}$ is distributed as $\{\boldsymbol{x}^{(1)},\dots,\boldsymbol{x}^{(m)}\} \sim \pi(\boldsymbol{x}|\boldsymbol{\theta},\boldsymbol{y})$, where the posterior distribution $\pi(\boldsymbol{x}|\boldsymbol{\theta},\boldsymbol{y})$ is conditioned on the hyper-parameters $\boldsymbol{\theta}$ and defined according to a Bayesian hierarchical model.

2.3 Bayesian Inference

In this section, we introduce the basics of Bayesian inference for an unknown parameter $\boldsymbol{x}$ given observed data

$$\boldsymbol{y} = \boldsymbol{A}\boldsymbol{x} + \boldsymbol{\eta}, \tag{2.4}$$

based on a linear forward model $\boldsymbol{A}$ and some additive noise $\boldsymbol{\eta}$. A more sophisticated Bayesian framework specifically applied to the previously introduced forward model and some simulated data will be developed in Section 3.2.

We can visualise the correlation structure between parameters as well as how distributions progress in a measurement process, using a hierarchically ordered directed acyclic graph (DAG), see Figure 2.3. Since any observational process naturally involves random noise, we include this in the DAG and classify the noise variance as a hyper-parameter within $\boldsymbol{\theta}$ [6]. Other hyper-parameters, to which we assign a hyper-prior distribution $\pi(\boldsymbol{\theta})$, may influence the parameters $\boldsymbol{x}$ either statistically (indicated by solid arrows), as in Figure 2.3, or deterministically (indicated by dashed arrows) if functional dependent on each other. Here we can incorporate prior knowledge of $\boldsymbol{\theta}$ and the parameter $\boldsymbol{x}$ by defining $\pi(\boldsymbol{\theta})$ and the prior distribution $\pi(\boldsymbol{x}|\boldsymbol{\theta})$ according to their receptive physical properties or functional dependences. This is one of the great strength of Bayesian modelling compared to e.g. regularisation, see section 2.4. Then the parameter $\boldsymbol{x}$ is mapped deterministically through the forward model onto the space of all measurables Ω. From this space, we statistically observe the actual data $\boldsymbol{y}$, which includes random (statistical) noise as mentioned above. The distribution of the data conditioned on the hyper-parameters $\boldsymbol{\theta}$ and the parameters $\boldsymbol{x}$ is called the likelihood function $\pi(\boldsymbol{y}|\boldsymbol{\theta}, \boldsymbol{x})$, which includes information about the measurement process through the forward model. Then given some observed data, we like to characterise the posterior distribution $\pi(\boldsymbol{\theta}, \boldsymbol{x}|\boldsymbol{y})$ of the underlying parameters and hyper-parameters by reversing the arrows in Figure 2.3.

The posterior distribution, our function of interest, is defined by Bayes' theorem

$$\pi(\boldsymbol{x}, \boldsymbol{\theta}|\boldsymbol{y}) = \frac{\pi(\boldsymbol{y}|\boldsymbol{x}, \boldsymbol{\theta})\pi(\boldsymbol{x}, \boldsymbol{\theta})}{\pi(\boldsymbol{y})}\,, \tag{2.5}$$

with the prior distribution $\pi(\boldsymbol{x}, \boldsymbol{\theta}) = \pi(\boldsymbol{x}|\boldsymbol{\theta})\pi(\boldsymbol{\theta})$ and the normalising constant $\pi(\boldsymbol{y})$. If the normalising constant is finite and non-zero we approximate the posterior distribution

$$\pi(\boldsymbol{x}, \boldsymbol{\theta}|\boldsymbol{y}) \propto \pi(\boldsymbol{y}|\boldsymbol{x}, \boldsymbol{\theta})\pi(\boldsymbol{x}, \boldsymbol{\theta})\,. \tag{2.6}$$

The expectation of any a function $h(\boldsymbol{x}_{\boldsymbol{\theta}})$, where $\boldsymbol{x}$ may depend on $\boldsymbol{\theta}$, is described as

$$\mathrm{E}_{\boldsymbol{x},\boldsymbol{\theta}|\boldsymbol{y}}[h(\boldsymbol{x}_{\boldsymbol{\theta}})] = \underbrace{\int\int h(\boldsymbol{x}_{\boldsymbol{\theta}})\,\pi(\boldsymbol{x}, \boldsymbol{\theta}|\boldsymbol{y})\,\mathrm{d}\boldsymbol{x}\,\mathrm{d}\boldsymbol{\theta}}_{\boldsymbol{\mu}_{\mathrm{int}}}\,, \tag{2.7}$$

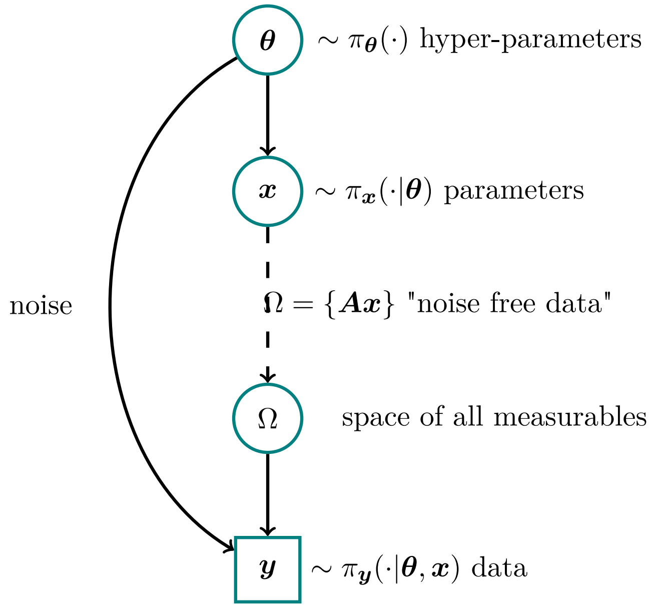

$\boldsymbol{\theta}$
$\sim \pi_{\boldsymbol{\theta}}(\cdot)$ hyper-parameters
$\boldsymbol{x}$
$\sim \pi_{\boldsymbol{x}}(\cdot|\boldsymbol{\theta})$ parameters
noise
$\Omega = \{\boldsymbol{Ax}\}$ "noise free data"
Ω
space of all measurables
$\boldsymbol{y}$
$\sim \pi_{\boldsymbol{y}}(\cdot|\boldsymbol{\theta}, \boldsymbol{x})$ data

Figure 2.3: The directed acyclic graph (DAG) for a linear inverse problem visualises statistical dependencies as solid line arrows and deterministic dependencies as dotted arrows. The parameters $\boldsymbol{x}$ have some statistical dependency of those hyper-parameters $\boldsymbol{\theta}$, which are distributed as $\pi(\boldsymbol{\theta})$. Then a parameter $\boldsymbol{x} \sim \pi_{\boldsymbol{x}}(\cdot|\boldsymbol{\theta})$ is mapped onto the space of all measurables $\boldsymbol{u} = \boldsymbol{Ax}$ deterministically through the linear forward model $\boldsymbol{A}$. From the space of all measurables we observe some data $\boldsymbol{y} = \boldsymbol{Ax} + \boldsymbol{\eta}$, statistically, so that $\boldsymbol{y} \sim \pi_{\boldsymbol{y}}(\cdot|\boldsymbol{\theta}, \boldsymbol{x})$, with naturally some random noise $\boldsymbol{\eta} \sim \pi_{\boldsymbol{\eta}}(\cdot|\boldsymbol{\theta})$.

which may be a high dimensional integral and computationally not feasible to solve. Therefore the unbiased [8] sample based Monte Carlo estimate

$$\mathrm{E}_{\boldsymbol{x},\boldsymbol{\theta}|\boldsymbol{y}}[h(\boldsymbol{x}_{\boldsymbol{\theta}})] \approx \underbrace{\frac{1}{N}\sum_{k=1}^{N} h(\boldsymbol{x}_{\boldsymbol{\theta}}^{(k)})}_{\boldsymbol{\mu}_{\text{samp}}}, \tag{2.8}$$

for large enough N (law of large numbers [7, Chapter 17]) is often used. Here, the samples $\{\boldsymbol{x}^{(k)}, \boldsymbol{\theta}^{(k)}\} \sim \pi_{\boldsymbol{x},\boldsymbol{\theta}}(\cdot|\boldsymbol{y})$, for $k = 1, \dots, N$, form a sample set $\mathcal{M} = \{(\boldsymbol{x}, \boldsymbol{\theta})^{(1)}, \dots, (\boldsymbol{x}, \boldsymbol{\theta})^{(N)}\}$. Furthermore, the central limit theorem states that the samples mean $\boldsymbol{\mu}_{samp}^{(i)}$, of independent samples sets $\mathcal{M}_i$ for $i = 1, \dots, n$ of any distribution, converge in distribution to a normal distribution so that

$$\sqrt{n}(\boldsymbol{\mu}_{\text{samp}}^{(i)} - \boldsymbol{\mu}_{\text{int}}) \xrightarrow{\mathcal{D}} \mathcal{N}(0, \sigma^2)[9], \tag{2.9}$$

and if $\sigma^2 < \infty$ the Monte Carlo error $\boldsymbol{\mu}_{\text{samp}}^{(i)} - \boldsymbol{\mu}_{\text{int}}$ is bounded.

Generating a representative sample set from the posterior distribution presents a significant challenge. This is also due to the strong correlations that often exist between the parameters and hyper-parameters, as discussed by Rue and Held in [10] and illustrated in Appendix A. If $\boldsymbol{x}$ can not be parametrised directly in terms of the hyper-parameters

$\boldsymbol{\theta}$, i.e., $\boldsymbol{x}(\boldsymbol{\theta})$, it is beneficial to factorise the posterior distribution as

$$\pi(\boldsymbol{x}, \boldsymbol{\theta}|\boldsymbol{y}) = \pi(\boldsymbol{x}|\boldsymbol{\theta}, \boldsymbol{y})\, \pi(\boldsymbol{\theta}|\boldsymbol{y}), \tag{2.10}$$

into the conditional posterior $\pi(\boldsymbol{x}|\boldsymbol{\theta}, \boldsymbol{y})$ over the latent field $\boldsymbol{x}$ and the marginal posterior $\pi(\boldsymbol{\theta}|\boldsymbol{y})$ over the hyper-parameters $\boldsymbol{\theta}$. This approach, known as the marginal and then conditional (MTC) method, is particularly advantageous when $\boldsymbol{x}$ is high-dimensional (e.g., $\boldsymbol{x} \in \mathbb{R}^n$ with $n = 45$), while $\boldsymbol{\theta}$ is low-dimensional (e.g., two-dimensional) and one can deterministically work out the marginal distribution. Applying the law of total expectation [11], Eq. (2.7) becomes

$$\mathbb{E}_{\boldsymbol{x}|\boldsymbol{y}}[h(\boldsymbol{x})] = \mathbb{E}_{\boldsymbol{\theta}|\boldsymbol{y}}\left[\mathbb{E}_{\boldsymbol{x}|\boldsymbol{\theta},\boldsymbol{y}}[h(\boldsymbol{x}_{\boldsymbol{\theta}})]\right] = \int \mathbb{E}_{\boldsymbol{x}|\boldsymbol{\theta},\boldsymbol{y}}\left[h(\boldsymbol{x}_{\boldsymbol{\theta}})\right]\, \pi(\boldsymbol{\theta}|\boldsymbol{y})\, \mathrm{d}\boldsymbol{\theta}, \tag{2.11}$$

where, in the case of a linear-Gaussian Bayesian hierarchical model, both the marginal distribution and the inner expectation $\mathbb{E}_{\boldsymbol{x}|\boldsymbol{\theta},\boldsymbol{y}}\left[h(\boldsymbol{x}_{\boldsymbol{\theta}})\right]$ are well defined.

Assuming Gaussian noise $\boldsymbol{\eta} \sim \mathcal{N}(0, \boldsymbol{\Sigma}(\boldsymbol{\theta}))$, we define a linear-Gaussian Bayesian hierarchical model [6]

$$\boldsymbol{y}|\boldsymbol{x}, \boldsymbol{\theta} \sim \mathcal{N}(\boldsymbol{A}\boldsymbol{x}, \boldsymbol{\Sigma}(\boldsymbol{\theta})) \tag{2.12a}$$
$$\boldsymbol{x}|\boldsymbol{\theta} \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{Q}^{-1}(\boldsymbol{\theta})) \tag{2.12b}$$
$$\boldsymbol{\theta} \sim \pi(\boldsymbol{\theta}), \tag{2.12c}$$

with a normally distributed likelihood $\pi(\boldsymbol{y}|\boldsymbol{x}, \boldsymbol{\theta})$ and prior distributions $\pi(\boldsymbol{x}|\boldsymbol{\theta})$ and $\pi(\boldsymbol{\theta})$, the noise covariance matrix $\boldsymbol{\Sigma}(\boldsymbol{\theta})$, the prior precision matrix $\boldsymbol{Q}(\boldsymbol{\theta})$ and the prior mean $\boldsymbol{\mu}$. This model enables efficient factorisation of the posterior distribution and application of the MTC method.

2.3.1 Marginal and conditional posterior distribution

For the linear-Gaussian Bayesian hierarchical model specified in Eq. 2.12, the marginal posterior distribution over the hyper-parameters is given by

$$\pi(\boldsymbol{\theta}|\boldsymbol{y}) = \int \pi(\boldsymbol{x}, \boldsymbol{\theta}|\boldsymbol{y})\, \mathrm{d}\boldsymbol{x} \tag{2.13}$$
$$\propto \sqrt{\frac{\det(\boldsymbol{\Sigma}^{-1})\, \det(\boldsymbol{Q})}{\det(\boldsymbol{Q} + \boldsymbol{A}^T\boldsymbol{\Sigma}^{-1}\boldsymbol{A})}} \times \exp\left[-\frac{1}{2}(\boldsymbol{y} - \boldsymbol{A}\boldsymbol{\mu})^T \boldsymbol{Q}_{\boldsymbol{\theta}|\boldsymbol{y}}(\boldsymbol{y} - \boldsymbol{A}\boldsymbol{\mu})\right] \pi(\boldsymbol{\theta})\,, \tag{2.14}$$

with

$$\boldsymbol{Q}_{\boldsymbol{\theta}|\boldsymbol{y}} = \boldsymbol{\Sigma}^{-1} - \boldsymbol{\Sigma}^{-1}\boldsymbol{A}\left(\boldsymbol{A}^T\boldsymbol{\Sigma}^{-1}\boldsymbol{A} + \boldsymbol{Q}\right)^{-1} \boldsymbol{A}^T\boldsymbol{\Sigma}^{-1}\,, \tag{2.15}$$

see [6, Lemma 2]. Conditioned on the hyper-parameters $\boldsymbol{\theta}$, we can draw samples from the normal conditional posterior distribution

$$\boldsymbol{x}|\boldsymbol{\theta},\boldsymbol{y} \sim \mathcal{N}\Big(\underbrace{\boldsymbol{\mu} + \left(\boldsymbol{A}^T\boldsymbol{\Sigma}^{-1}\boldsymbol{A} + \boldsymbol{Q}\right)^{-1}\boldsymbol{A}^T\boldsymbol{\Sigma}^{-1}(\boldsymbol{y} - \boldsymbol{A}\boldsymbol{\mu})}_{\boldsymbol{\mu}_{\boldsymbol{x}|\boldsymbol{\theta},\boldsymbol{y}}}, \underbrace{\left(\boldsymbol{A}^T\boldsymbol{\Sigma}^{-1}\boldsymbol{A} + \boldsymbol{Q}\right)^{-1}}_{\boldsymbol{\Sigma}_{\boldsymbol{x}|\boldsymbol{y},\boldsymbol{\theta}}}\Big), \tag{2.16}$$

using the Randomise-then-Optimise (RTO) method (see Section 2.5.2), or compute weighted expectations, as in Eq. 2.11, of the conditional mean and covariance matrix, where the weights are given by $\pi(\boldsymbol{\theta}|\boldsymbol{y})$. Note that both the noise covariance $\boldsymbol{\Sigma} = \boldsymbol{\Sigma}(\boldsymbol{\theta})$ and the prior precision matrix $\boldsymbol{Q} = \boldsymbol{Q}(\boldsymbol{\theta})$ depend on the hyper-parameters $\boldsymbol{\theta}$.

2.4 Regularisation

Another method for obtaining a solution to the linear inverse problem in Eq. 2.4 is regularisation. In this approach, we seek a solution $\boldsymbol{x}_\lambda$ that minimises both the data misfit norm and a regularisation semi-norm, as described in [6]. Here we focus on a regularisation semi-norm for the case of Tikhonov regularisation [12, 13], which is closest to a linear-Gaussian hierarchical Bayesian model, as introduced in Eq. 2.12.

Given a parameter vector $\boldsymbol{x}$, a linear forward model matrix $\boldsymbol{A}$, and data $\boldsymbol{y}$, the data misfit norm

$$\|\boldsymbol{y} - \boldsymbol{A}\boldsymbol{x}\| \tag{2.17}$$

quantifies how well the noise-free data $\boldsymbol{A}\boldsymbol{x}$ matches the observed data. The regularisation semi-norm

$$\lambda \|\boldsymbol{T}\boldsymbol{x}\| \tag{2.18}$$

penalises the solution according to the regularisation operator $\boldsymbol{T}$ and the regularisation parameter $\lambda > 0$. For a fixed λ, the regularised solution

$$\boldsymbol{x}_\lambda = \arg\min_{\boldsymbol{x}} \|\boldsymbol{y} - \boldsymbol{A}\boldsymbol{x}\|^2 + \lambda \|\boldsymbol{T}\boldsymbol{x}\|^2 \tag{2.19}$$

is obtained by taking the derivative with respect to $\boldsymbol{x}$ of the objective function:

$$\begin{aligned} & \nabla_{\boldsymbol{x}} \left\{(\boldsymbol{y} - \boldsymbol{A}\boldsymbol{x})^T(\boldsymbol{y} - \boldsymbol{A}\boldsymbol{x}) + \lambda\boldsymbol{x}^T\boldsymbol{T}^T\boldsymbol{T}\boldsymbol{x}\right\} = 0 && (2.20)\\ \iff \quad & \nabla_{\boldsymbol{x}} \left\{\boldsymbol{y}^T\boldsymbol{y} + \boldsymbol{x}^T\boldsymbol{A}^T\boldsymbol{A}\boldsymbol{x} - 2\boldsymbol{y}^T\boldsymbol{A}\boldsymbol{x} + \lambda\boldsymbol{x}^T\boldsymbol{T}^T\boldsymbol{T}\boldsymbol{x}\right\} = 0 && (2.21)\\ \iff \quad & 2\boldsymbol{A}^T\boldsymbol{A}\boldsymbol{x} - 2\boldsymbol{A}^T\boldsymbol{y} + 2\lambda\boldsymbol{T}^T\boldsymbol{T}\boldsymbol{x} = 0. && (2.22) \end{aligned}$$

Solving this equation yields the regularised solution

$$\boldsymbol{x}_\lambda = (\boldsymbol{A}^T\boldsymbol{A} + \lambda\boldsymbol{L})^{-1}\boldsymbol{A}^T\boldsymbol{y}\,, \tag{2.23}$$

where we define $\boldsymbol{L} := \boldsymbol{T}^T\boldsymbol{T}$, which typically represents a discrete approximation of a derivative operator [13].

In practice, $\boldsymbol{x}_\lambda$ is computed for a range of λ-values and evaluated based on the trade-off between the data misfit and the regularisation norm. The optimal value of λ is often chosen as the point of maximum curvature on the so-called L-curve [14], which we plot in Section 3.2.

2.5 Sampling Methods

In this section we present the sampling methods used in this thesis and show how these methods draw samples $\mathcal{M} = \{(\boldsymbol{x}, \boldsymbol{\theta})^{(1)}, \dots, (\boldsymbol{x}, \boldsymbol{\theta})^{(k)}, \dots, (\boldsymbol{x}, \boldsymbol{\theta})^{(N)}\} \sim \pi(\boldsymbol{x}, \boldsymbol{\theta}|\boldsymbol{y})$ from the desired target distribution, so that we can apply sample-based estimates as in Eq. 2.8. Here, $\mathcal{M}$ denotes a Markov chain, where each new sample $(\boldsymbol{x}, \boldsymbol{\theta})^{(k)}$ is only affected by the previous one, $(\boldsymbol{x}, \boldsymbol{\theta})^{(k-1)}$. Markov chain Monte Carlo (MCMC) methods generate such a chain $\mathcal{M}$ using random (Monte Carlo) proposals $(\boldsymbol{x}, \boldsymbol{\theta})^{(k)} \sim q(\cdot|(\boldsymbol{x}, \boldsymbol{\theta})^{(k-1)})$ according to a proposal distribution conditioned on the previous sample (Markov), where ergodicity of $\mathcal{M}$ is a sufficient criterion for using sample-based estimates [8, 13].

The ergodicity theorem in [13] states that, if a Markov chain $\mathcal{M}$ is aperiodic, irreducible, and reversible, then it converges to a unique stationary equilibrium distribution. In other words, if the chain can reach any state from any other state (irreducibility), is not stuck in periodic cycles (aperiodicity), and is reversible (detailed balance condition [13]), then it will converge to the desired target distribution $\pi(\boldsymbol{x}, \boldsymbol{\theta}|\boldsymbol{y})$. In practice, one can inspect the trace $\pi(\boldsymbol{x}^{(k)}, \boldsymbol{\theta}^{(k)}|\boldsymbol{y})$ for $k = 1, \dots, N$ and visually assess convergence and mixing properties of the chain to evaluate ergodicity. The sampling methods used in this thesis possess proven ergodic properties, and we therefore refer the reader to the corresponding literature for further details.

2.5.1 Metropolis- within Gibbs sampling

As introduced in Section 2.3.1, when using the MTC method we sample separately from $\pi(\boldsymbol{\theta}|\boldsymbol{y})$ and $\pi(\boldsymbol{x}|\boldsymbol{\theta}, \boldsymbol{y})$. To sample from $\pi(\boldsymbol{\theta}|\boldsymbol{y})$, we use a Metropolis-within-Gibbs sampler as described in [6]. In this thesis, the sampler is applied to the two-dimensional case only, with $\boldsymbol{\theta} = (\theta_1, \theta_2)$, where we perform a Metropolis step in the θ_1 direction and a Gibbs step in the θ_2 direction. Ergodicity for this approach is proven in [15].

The Metropolis-within-Gibbs algorithm begins with an initial guess $\boldsymbol{\theta}^{(t)}$ at $t = 0$. We then propose a new sample $\theta_1 \sim q(\theta_1|\theta_1^{(t-1)})$, conditioned on the previous state, using a symmetric proposal distribution $q(\theta_1|\theta_1^{(t-1)}) = q(\theta_1^{(t-1)}|\theta_1)$, which is a special case of the Metropolis-Hastings algorithm [15]. We accept and set $\theta_1^{(t)} = \theta_1$ with the acceptance probability

$$\alpha(\theta_1|\theta_1^{(t-1)}) = \min\left\{1, \frac{\pi(\theta_1|\theta_2^{(t-1)}, \boldsymbol{y})\, \cancel{q(\theta_1^{(t-1)}|\theta_1)}}{\pi(\theta_1^{(t-1)}|\theta_2^{(t-1)}, \boldsymbol{y})\, \cancel{q(\theta_1|\theta_1^{(t-1)})}}\right\} \tag{2.24}$$

or reject and keep $\theta_1^{(t)} = \theta_1^{(t-1)}$, which we do by comparing α to a uniform random number $u \sim \mathcal{U}(0,1)$.

Next, we perform a Gibbs step in the θ_2 direction, where Gibbs sampling is again a special case of the Metropolis-Hastings algorithm with acceptance probability equal to one, and draw the next sample $\theta_2^{(t)} \sim \pi(\cdot|\theta_1^{(t)}, \boldsymbol{y})$, conditioned on the current value $\theta_1^{(t)}$.

We repeat this procedure N' times and ensure convergence independently of the initial sample (irreducibility) by discarding the initial $N_{\text{burn-in}}$ samples after a so-called burn-in period, resulting in a Markov chain of length $N = N' - N_{\text{burn-in}}$.

Algorithm 1: Metropolis within Gibbs

```
Initialize and suppose two dimensional vector $\boldsymbol{\theta}^{(0)} = (\theta_1^{(0)}, \theta_2^{(0)})$
for $k = 1, \dots, N'$ do
    Propose $\theta_1 \sim q(\cdot|\theta_1^{(t-1)}) = q(\theta_1^{(t-1)}|\cdot)$
    Compute
        $\alpha(\theta_1|\theta_1^{(t-1)}) = \min\left\{1, \frac{\pi(\theta_1|\theta_2^{(t-1)}, \boldsymbol{y})\cancel{q(\theta_1^{(t-1)}|\theta_1)}}{\pi(\theta_1^{(t-1)}|\theta_2^{(t-1)}, \boldsymbol{y})\cancel{q(\theta_1|\theta_1^{(t-1)})}}\right\}$
    Draw $u \sim \mathcal{U}(0,1)$
    if $\alpha \geq u$ then
        Accept and set $\theta_1^{(t)} = \theta_1$
    else
        Reject and keep $\theta_1^{(t)} = \theta_1^{(t-1)}$
    end if
    Draw $\theta_2^{(t)} \sim \pi(\cdot|\theta_1^{(t)}, \boldsymbol{y})$
end for
Output: $\boldsymbol{\theta}^{(0)}, \dots, \boldsymbol{\theta}^{(k)}, \dots, \boldsymbol{\theta}^{(N)} \sim \pi(\boldsymbol{\theta}|\boldsymbol{y})$
```

2.5.2 Draw a sample from a multivariate normal distribution

As part of the MTC scheme, we only draw samples from the conditional distribution $\pi(\boldsymbol{x}|\boldsymbol{\theta}, \boldsymbol{y})$ after sampling from the marginal posterior $\pi(\boldsymbol{\theta}|\boldsymbol{y})$. For linear-Gaussian Bayesian hierarchical models, samples from the multivariate normal distribution $\pi(\boldsymbol{x}|\boldsymbol{\theta}, \boldsymbol{y})$ can be efficiently generated using the Randomise-then-Optimise (RTO) method [16].

The full conditional distribution can be rewritten as

$$\pi(\boldsymbol{x}|\boldsymbol{y}, \boldsymbol{\theta}) \propto \pi(\boldsymbol{y}|\boldsymbol{x}, \boldsymbol{\theta})\, \pi(\boldsymbol{x}|\boldsymbol{\theta}) \tag{2.25}$$

$$= \exp\left(-\left\|\hat{\boldsymbol{A}}\boldsymbol{x} - \hat{\boldsymbol{y}}\right\|^2\right), \tag{2.26}$$

where

$$\hat{\boldsymbol{A}} = \begin{bmatrix} \boldsymbol{\Sigma}^{-1/2}(\boldsymbol{\theta})\boldsymbol{A} \\ \boldsymbol{Q}^{1/2}(\boldsymbol{\theta}) \end{bmatrix}, \quad \hat{\boldsymbol{y}} = \begin{bmatrix} \boldsymbol{\Sigma}^{-1/2}(\boldsymbol{\theta})\boldsymbol{y} \\ \boldsymbol{Q}^{1/2}(\boldsymbol{\theta})\boldsymbol{\mu} \end{bmatrix} \quad [17]. \tag{2.27}$$

A sample $\boldsymbol{x}_i$ can be computed by minimising the following equation with respect to $\hat{\boldsymbol{x}}$:

$$\boldsymbol{x}_i = \arg\min_{\hat{\boldsymbol{x}}} \| \hat{\boldsymbol{A}}\hat{\boldsymbol{x}} - (\hat{\boldsymbol{y}} + \boldsymbol{b})\|^2, \quad \boldsymbol{b} \sim \mathcal{N}(\boldsymbol{0}, \mathbf{I}) \,, \tag{2.28}$$

where we add a randomised perturbation $\boldsymbol{b}$. Similar to Section 2.4, this expression can be rewritten as

$$\left(\boldsymbol{A}^T \boldsymbol{\Sigma}^{-1}(\boldsymbol{\theta})\boldsymbol{A} + \boldsymbol{Q}(\boldsymbol{\theta}) \right) \boldsymbol{x}_i = \boldsymbol{A}^T \boldsymbol{\Sigma}^{-1}(\boldsymbol{\theta})\boldsymbol{y} + \boldsymbol{Q}(\boldsymbol{\theta})\boldsymbol{\mu} + \boldsymbol{v}_1 + \boldsymbol{v}_2, \tag{2.29}$$

where the term $-\hat{\boldsymbol{A}}^T\boldsymbol{b}$ is decomposed as $\boldsymbol{v}_1 + \boldsymbol{v}_2$, with $\boldsymbol{v}_1 \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{A}^T\boldsymbol{\Sigma}^{-1}(\boldsymbol{\theta})\boldsymbol{A})$ and $\boldsymbol{v}_2 \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{Q}(\boldsymbol{\theta}))$, representing independent Gaussian random variables [6, 16].

If the Markov chain over the marginal posterior $\pi(\boldsymbol{\theta}|\boldsymbol{y})$ is ergodic, and the conditional samples $\boldsymbol{x}^{(k)} \sim \pi(\boldsymbol{x}|\boldsymbol{\theta}^{(k)}, \boldsymbol{y})$ are drawn independently, then the resulting joint chain $\{(\boldsymbol{x}, \boldsymbol{\theta})^{(1)}, \dots, (\boldsymbol{x}, \boldsymbol{\theta})^{(N)}\} \sim \pi(\boldsymbol{x}, \boldsymbol{\theta}|\boldsymbol{y})$ is also ergodic [18].

2.5.3 t-walk sampler as black box

If the parameters $\boldsymbol{x}$ are functionally dependent on the hyper-parameters $\boldsymbol{\theta}$, i.e., $\boldsymbol{x} = \boldsymbol{x}(\boldsymbol{\theta})$, we can sample directly from the marginal posterior $\pi(\boldsymbol{\theta}|\boldsymbol{y})$ using the t-walk algorithm by Christen and Fox [19]. The t-walk is employed as a black-box sampler, requiring only the specification of the number of samples, burn-in period, support region, and the sampling distribution. Convergence to the target distribution is guaranteed by construction of the algorithm.

2.6 Numerical Approxiamtion Methods - Tensor Train

First, we provide a short overview of probability spaces and their associated measures, as a foundation for deriving marginal probability distribution, and then we give a brief introduction to the tensor train format.

Assume that the triple $(\Omega, \mathcal{F}, \mathbb{P})$ defines a probability space, where Ω denotes the complete sample space, $\mathcal{F}$ is a σ-algebra consisting of a collection of countable subsets $\{A_n\}_{n \in \mathbb{N}}$ with $A_n \subseteq \Omega$, and $\mathbb{P}$ is a probability measure defined on $\mathcal{F}$. The formal conditions for $\mathbb{P}$ to be a probability measure, and for $\mathcal{F}$ to be a σ-algebra over Ω, are given in Appendix B. We denote

$$\mathbb{P}(A) = \int_A \mathrm{d}\mathbb{P} \tag{2.30}$$

as the probability of an event $A \in \mathcal{F}$. By applying the Radon-Nikodym theorem [20], we can change variables

$$\mathbb{P}(A) = \int_A \frac{\mathrm{d}\mathbb{P}}{\mathrm{d}x}\,\mathrm{d}x = \int_A \pi(x)\,\mathrm{d}x, \tag{2.31}$$

where $\mathrm{d}x$ is a reference measure on the same probability space, commonly referred to as the Lebesgue measure. The Radon-Nikodym derivative $\frac{\mathrm{d}\mathbb{P}}{\mathrm{d}x}$ of $\mathbb{P}$ with respect to x, and is often interpreted as the probability density function (PDF) $\pi(x)$. Thus, we say that $\mathbb{P}$ has a density $\pi(x)$ with respect to x [21, Chapter 10].

Now, let $X : \Omega \longrightarrow \mathbb{R}^d$ be a d-dimensional random variable mapping from the probability space $(\Omega, \mathcal{F}, \mathbb{P})$ to the measurable space $(\mathbb{R}^d, \mathcal{X})$, where $\mathcal{X}$ is a collection of subsets in $\mathbb{R}^d$. Then the associated PDF $\pi(x)$, is a joint density of X, induced by the probability measure on Ω [20, 22]. As by Cui et al. [23], we can define the parameter space as the Cartesian product $\mathcal{X} = \mathcal{X}_1 \times \mathcal{X}_2 \times \cdots \times \mathcal{X}_d$ with $x_k \in \mathcal{X}_k \subseteq \mathbb{R}$ and $x = (x_1, \ldots, x_k, \ldots, x_d)$. The marginal density function for the k-th component is then given by

$$f_{X_k}(x_k) = \int_{\mathcal{X}_1} \cdots \int_{\mathcal{X}_d} \lambda(x)\,\pi(x)\,\mathrm{d}x_1 \cdots \mathrm{d}x_{k-1}\,\mathrm{d}x_{k+1} \cdots \mathrm{d}x_d, \tag{2.32}$$

where we integrate over all dimensions except the k-th. Here, we introduce a weight function $\lambda(x)$ [24], which can be useful for quadrature rules??. Cui et al. [23] refer to $\lambda(x)$ as a "product-form Lebesgue-measurable weighting function" and define it as

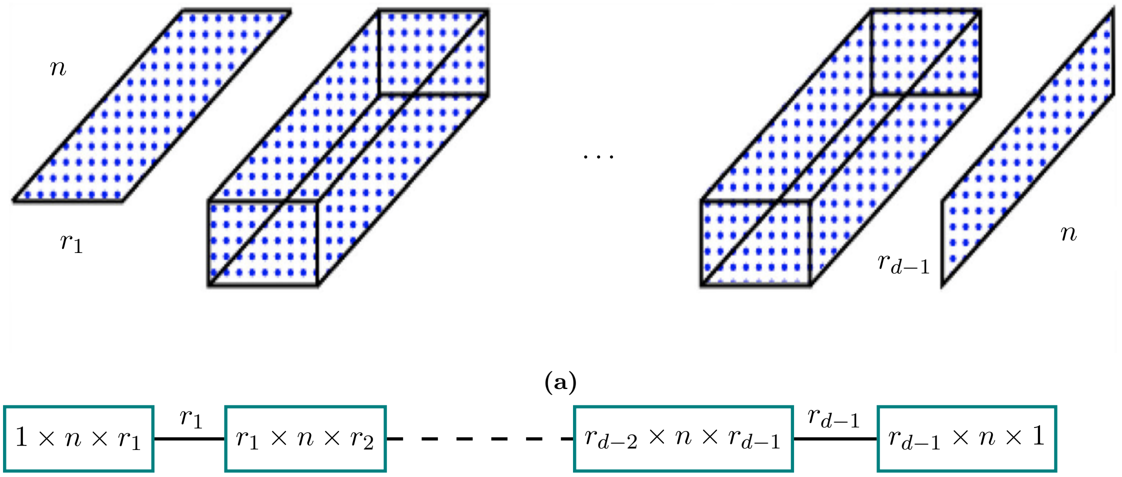
$$\lambda(\mathcal{X}) = \prod_{i=1}^{d} \lambda_i(\mathcal{X}_i), \quad \text{where} \quad \lambda_i(\mathcal{X}_i) = \int_{\mathcal{X}_i} \lambda_i(x_i)\,\mathrm{d}x_i.$$

Using the tensor train (TT) format, we can efficiently approximate a d-dimensional function $\pi(x)$ and compute marginal probability distributions at low computational cost. To do so, we first define a d-dimensional discrete univariate grid over the parameter space $\mathcal{X}$, with n grid points in each dimension. In the tensor train format we can represent the function over this d-dimensional grid as a product train of rank-2 to rank-3 tensors, which we call TT-cores. More specifically each core $\pi_k \in \mathbb{R}^{r_{k-1} \times n \times r_k}$ has ranks r_{k-1} and r_k, for $k = 1, \ldots, d$, connecting it with its neighbouring cores, as illustrated in Figure 2.4. For the first and last cores, the outer ranks are set to $r_0 = r_d = 1$. This enables us to write the value $\pi(x)$, for a fixed point $x = (x_1, \ldots, x_d)$ on the grid, as a sequence of matrix multiplications

$$\pi_1(x_1)\pi_2(x_2) \cdots \pi_d(x_d) = \pi(x) \in \mathbb{R},$$

where each core $\pi_k(x_k)$, becomes a matrix of size $r_{k-1} \times r_k$. Consequently, with a tensor train approximation, the marginal target function

$$\begin{aligned} f_{X_k}(x_k) = \frac{1}{z}\Big| &\left(\int_{\mathbb{R}} \lambda_1(x_1)\pi_1(x_1)\,\mathrm{d}x_1\right) \cdots \left(\int_{\mathbb{R}} \lambda_{k-1}(x_{k-1})\pi_{k-1}(x_{k-1})\,\mathrm{d}x_{k-1}\right) \\ &\lambda_k(x_k)\pi_k(x_k) \\ &\left(\int_{\mathbb{R}} \lambda_{k+1}(x_{k+1})\pi_{k+1}(x_{k+1})\,\mathrm{d}x_{k+1}\right) \cdots \left(\int_{\mathbb{R}} \lambda_d(x_d)\pi_d(x_d)\,\mathrm{d}x_d\right)\Big| \end{aligned} \tag{2.33}$$


n
r_1
$\cdots$
r_{d-1}
n
$1 \times n \times r_1$
r_1
$r_1 \times n \times r_2$
$r_{d-2} \times n \times r_{d-1}$
r_{d-1}
$r_{d-1} \times n \times 1$

(a)

(b)

Figure 2.4: Here, we visualise the tensor train cores as two- and three-dimensional matrices. Each core has a length n, corresponding to the number of grid points in one dimension, and the cores are connected through ranks r_k. More specifically, a core π_k has dimensions $r_{k-1} \times n \times r_k$, with outer ranks $r_0 = r_d = 1$. Figure (a) is adapted from [25].

is computed by integrating over all TT cores except π_k, as in [26], including a normalisation constant z [23].

In practice, tensor train approximations may suffer from numerical instability, particularly because it is not advantageous to approximate the target function $\pi(x)$ in for example, the logarithmic space. To address this, we follow the notation and procedure of Cui et al. [23] and instead approximate the square root of the probability density

$$\sqrt{\pi(x)} \approx g(x) = \boldsymbol{G}_1(x_1), \ldots, \boldsymbol{G}_k(x_k), \ldots, \boldsymbol{G}_d(x_d). \tag{2.34}$$

Here, each TT-core is given by

$$G_k^{(\alpha_{k-1},\alpha_k)}(x_k) = \sum_{i=1}^{n_k} \phi_k^{(i)}(x_k) \boldsymbol{A}_k[\alpha_{k-1}, i, \alpha_k], \quad k = 1, \ldots, d, \tag{2.35}$$

where $\boldsymbol{A}_k \in \mathbb{R}^{r_{k-1} \times n_k \times r_k}$ is the k-th coefficient tensor and $\{\phi_k^{(i)}(x_k)\}_{i=1}^{n_k}$ are the basis functions corresponding to the k-th coordinate. The approximated density is written as:

$$\pi(x) \approx \gamma' + g^2(x), \tag{2.36}$$

where γ' is a small positive constant added to ensure positivity and is chosen such that

$$\gamma' \leq \frac{1}{\lambda(\mathcal{X})} \|g - \sqrt{\pi}\|_2^2. \tag{2.37}$$

This leads to the normalised target function

$$f_X(x) = \frac{1}{z}\lambda(x)\pi(x) = \frac{1}{z}\left(\lambda(x)\gamma' + \lambda(x)g^2(x)\right), \tag{2.38}$$

where z is the normalisation constant. Given the tensor train approximation of $\sqrt{\pi}$, the marginal function $f_{X_k}(x_k)$ can be expressed as

$$\begin{aligned} f_{X_k}(x_k) = \frac{1}{z}\Bigg(&\gamma' \prod_{i=1}^{k-1} \boldsymbol{\lambda}_i(\mathcal{X}_i) \prod_{i=k+1}^{d} \boldsymbol{\lambda}_i(\mathcal{X}_i) \\ &+ \left(\int_{\mathbb{R}} \boldsymbol{\lambda}_1(x_1)\boldsymbol{G}_1^2(x_1)\,\mathrm{d}x_1\right)\cdots\left(\int_{\mathbb{R}} \boldsymbol{\lambda}_{k-1}(x_{k-1})\boldsymbol{G}_{k-1}^2(x_{k-1})\,\mathrm{d}x_{k-1}\right) \\ &\boldsymbol{\lambda}_k(x_k)\boldsymbol{G}_k^2(x_k) \\ &\left(\int_{\mathbb{R}} \boldsymbol{\lambda}_{k+1}(x_{k+1})\boldsymbol{G}_{k+1}^2(x_{k+1})\,\mathrm{d}x_{k+1}\right)\cdots\left(\int_{\mathbb{R}} \boldsymbol{\lambda}_d(x_d)\boldsymbol{G}_d^2(x_d)\,\mathrm{d}x_d\right)\Bigg). \end{aligned} \tag{2.39}$$

To compute these marginals efficiently, one can use a procedure similar to left and right orthogonalisation of TT-cores [27]. For this, we define the mass matrix $\boldsymbol{M}_k \in \mathbb{R}^{n_k \times n_k}$ as

$$\boldsymbol{M}_k[i,j] = \int_{\mathcal{X}_k} \phi_k^{(i)}(x_k)\phi_k^{(j)}(x_k)\boldsymbol{\lambda}(x_k)\,\mathrm{d}x_k, \quad i,j = 1,\ldots,n_k, \tag{2.40}$$

where $\{\phi_k^{(i)}(x_k)\}_{i=1}^{n_k}$ denotes the set of basis functions for the k-th coordinate.

2.6.1 Marginal Functions

We compute the marginal functions using two procedures, referred to as backward marginalisation [23] and forward marginalisation. The backward marginalisation provides us with the coefficient matrices $\boldsymbol{B}_k$, while the forward marginalisation gives the coefficient matrices $\boldsymbol{B}_{\mathrm{pre},n}$. These matrices enable the efficient evaluation of marginal functions, similar to [23]. The proposition used to compute $\boldsymbol{B}_k$, stated in Proposition 1, is adapted directly from [23].

Proposition 1 (Backward Marginalisation)**:** Starting with the last coordinate $k = d$, we set $\boldsymbol{B}_d = \boldsymbol{A}_d$. The following procedure can be used to obtain the coefficient tensor $\boldsymbol{B}_{k-1} \in \mathbb{R}^{r_{k-2}\times n_{k-1}\times r_{k-1}}$, which we need for defining the marginal function $f_{X_k}(x_k)$:

1. Use the Cholesky decomposition of the mass matrix, $\boldsymbol{L}_k\boldsymbol{L}_k^\top = \boldsymbol{M}_k \in \mathbb{R}^{n_k\times n_k}$, to construct a tensor $\boldsymbol{C}_k \in \mathbb{R}^{r_{k-1}\times n_k\times r_k}$:

$$\boldsymbol{C}_k[\alpha_{k-1}, \tau, l_k] = \sum_{i=1}^{n_k} \boldsymbol{B}_k[\alpha_{k-1}, i, l_k]\boldsymbol{L}_k[i, \tau]. \tag{2.41}$$

2. Unfold $\boldsymbol{C}_k$ along the first coordinate and compute the thin QR decomposition, so that $\boldsymbol{C}_k^{(R)} \in \mathbb{R}^{r_{k-1}\times(n_k r_k)}$:

$$\boldsymbol{Q}_k\boldsymbol{R}_k = (\boldsymbol{C}_k^{(R)})^\top. \tag{2.42}$$

3. Compute the new coefficient tensor:

$$\boldsymbol{B}_{k-1}[\alpha_{k-2}, i, l_{k-1}] = \sum_{\alpha_{k-1}=1}^{r_{k-1}} \boldsymbol{A}_{k-1}[\alpha_{k-2}, i, \alpha_{k-1}]\boldsymbol{R}_k[l_{k-1}, \alpha_{k-1}]. \tag{2.43}$$

Proposition 2 (Forward Marginalisation)**:** Starting with the first coordinate $k = 1$, we set $\boldsymbol{B}_{\text{pre},1} = \boldsymbol{A}_1$. The following procedure can be used to obtain the coefficient tensor $\boldsymbol{B}_{\text{pre},k+1} \in \mathbb{R}^{r_k\times n_{k+1}\times r_{k+1}}$ for defining the marginal function $f_{X_k}(x_k)$:

1. Use the Cholesky decomposition of the mass matrix, $\boldsymbol{L}_k\boldsymbol{L}_k^\top = \boldsymbol{M}_k \in \mathbb{R}^{n_k\times n_k}$, to construct a tensor $\boldsymbol{C}_k \in \mathbb{R}^{r_{k-1}\times n_k\times r_k}$:

$$\boldsymbol{C}_{\text{pre},k}[\alpha_{k-1}, \tau, l_k] = \sum_{i=1}^{n_k} \boldsymbol{L}_k[i, \tau]\boldsymbol{B}_{\text{pre},k}[\alpha_{k-1}, i, l_k]. \tag{2.44}$$

2. Unfold $\boldsymbol{C}_{pre,k}$ along the first coordinate and compute the thin QR decomposition, so that $\boldsymbol{C}_{\text{pre},k}^{(R)} \in \mathbb{R}^{(r_{k-1}n_k)\times r_k}$:

$$\boldsymbol{Q}_{pre,k}\boldsymbol{R}_{\text{pre},k} = (\boldsymbol{C}_{\text{pre},k}^{(R)}). \tag{2.45}$$

3. Compute the new coefficient tensor $\boldsymbol{B}_{\text{pre},k+1} \in \mathbb{R}^{r_{k-1}\times n_k\times r_k}$:

$$\boldsymbol{B}_{\text{pre},k+1}[l_{k+1}, i, \alpha_{k+1}] = \sum_{\alpha_k=1}^{r_k} \boldsymbol{R}_{\text{pre},k}[l_{k+1}, \alpha_k]\boldsymbol{A}_{k+1}[\alpha_k, i, \alpha_{k+1}]. \tag{2.46}$$

After computing the coefficient tensors $\boldsymbol{B}_{\text{pre},k+1}$ as in Prop. 2 and $\boldsymbol{B}_{k+1}$ from Prop. 1, the marginal PDF of k-th dimension can be expressed as

$$f_{X_k}(x_k) = \frac{1}{z}\left(\gamma'\prod_{i=1}^{k-1}\boldsymbol{\lambda}_i(X_i)\prod_{i=k+1}^{d}\boldsymbol{\lambda}_i(X_i) + \sum_{l_{k-1}=1}^{r_{k-1}}\sum_{l_k=1}^{r_k}\left(\sum_{i=1}^{n}\phi_k^{(i)}(x_k)\boldsymbol{D}_k[l_{k-1}, i, l_k]\right)^2\right)\boldsymbol{\lambda}_k(x_k), \tag{2.47}$$

where $\boldsymbol{D}_k \in \mathbb{R}^{r_{k-1}\times n\times r_k}$ and $\boldsymbol{R}_{\text{pre},k-1} \in \mathbb{R}^{r_{k-1}\times r_{k-1}}$ and $\boldsymbol{B}_k \in \mathbb{R}^{r_{k-1}\times n\times r_k}$

$$\boldsymbol{D}_k[l_{k-1}, i, l_k] = \sum_{\alpha_{k-1}=1}^{r_{k-1}} \boldsymbol{R}_{\text{pre},k-1}[l_{k-1}, \alpha_{k-1}]\boldsymbol{B}_k[\alpha_{k-1}, i, l_k]. \tag{2.48}$$

For the first dimension, $f_{X_1}(x_1)$ can be expressed as

$$f_{X_1}(x_1) = \frac{1}{z}\left(\gamma' \prod_{i=2}^{d} \boldsymbol{\lambda}_i(\mathcal{X}_i) + \sum_{l_1=1}^{r_1}\left(\sum_{i=1}^{n} \phi_1^{(i)}(x_1)\boldsymbol{D}_1[i, l_1]\right)^2\right)\boldsymbol{\lambda}_1(x_1), \tag{2.49}$$

where $\boldsymbol{D}_1[i, l_1] = \boldsymbol{B}_1[\alpha_0, i, l_1]$ and $\alpha_0 = 1$, and similarly in the last dimension

$$f_{X_d}(x_d) = \frac{1}{z}\left(\gamma' \prod_{i=1}^{d-1} \boldsymbol{\lambda}_i(\mathcal{X}_i) + \sum_{l_{n-1}=1}^{r_{d-1}}\left(\sum_{i=1}^{n} \phi_1^{(i)}(x_1)\boldsymbol{D}_d[l_{n-1}, i]\right)^2\right)\boldsymbol{\lambda}_d(x_d), \tag{2.50}$$

where $\boldsymbol{D}_d[l_{n-1}, i] = \boldsymbol{B}_{\text{pre},d}[l_{n-1}, i, \alpha_{n+1}]$ and $\alpha_{d+1} = 1$.

3 Results

3.1 Simulate data

3.2 Develop a hierarchical Bayesian model

3.2.1 prior modelling

3.3 affine map with posterior ozone

3.3.1 sampling vs TT

3.4 posterior ozone vs regularisation

3.4.1 sampling vs TT

3.5 posterior pressure and temperature

3.5.1 sampling vs TT

Appendices

A
Correlation Structure

In the book Gaussian Markov Random Fields [10], Rue and Held demonstrate that a strong correlation between the hyper-parameter μ and the latent field $\boldsymbol{x}$ can significantly slow down convergence when using samplers, in particularly Gibbs samplers. They consider the hierarchical model

$$\mu \sim \mathcal{N}(0, 1) \tag{A.1a}$$

$$\boldsymbol{x}|\mu \sim \mathcal{N}(\mu\mathbf{1}, \boldsymbol{Q}^{-1}), \tag{A.1b}$$

and apply a Gibbs sampler based on the full conditional distributions

$$\mu^{(k)}|\boldsymbol{x}^{(k)} \sim \mathcal{N}\left(\frac{\mathbf{1}^T\boldsymbol{Q}\boldsymbol{x}^{(k-1)}}{1+\mathbf{1}^T\boldsymbol{Q}\mathbf{1}}, \left(1+\mathbf{1}^T\boldsymbol{Q}\mathbf{1}\right)^{-1}\right) \tag{A.2}$$

$$\boldsymbol{x}^{(k)}|\mu^{(k)} \sim \mathcal{N}(\mu^{(k)}\mathbf{1}, \boldsymbol{Q}^{-1}). \tag{A.3}$$

As illustrated in Figure A.1, when the sampler is restricted to steps only in the μ-direction (horizontal axis) or the $\boldsymbol{x}$-direction (vertical axis), it requires many iterations to adequately explore the parameter space. This inefficiency arises from the high correlation between μ and $\boldsymbol{x}$, visible in Figure A.1 as a 'squeeze' of the distribution.

A solution to the slow mixing problem is to update $(\mu, \boldsymbol{x})$ jointly. Since here μ is one dimensional, effectively only marginal density of μ is needed.

$$\mu^\star \sim q(\mu^\star|\mu^{(k-1)}) \tag{A.4}$$

$$\boldsymbol{x}^{(k)}|\mu^\star \sim \mathcal{N}(\mu^\star\mathbf{1}, \boldsymbol{Q}^{-1}) \tag{A.5}$$

With a simple MCMC algorithm targeting μ one can explore the sample space efficiently and only draw a corresponding sample for $\boldsymbol{x}$ from its full conditional once, for instance, the proposal $\mu^\star$ has been accepted.

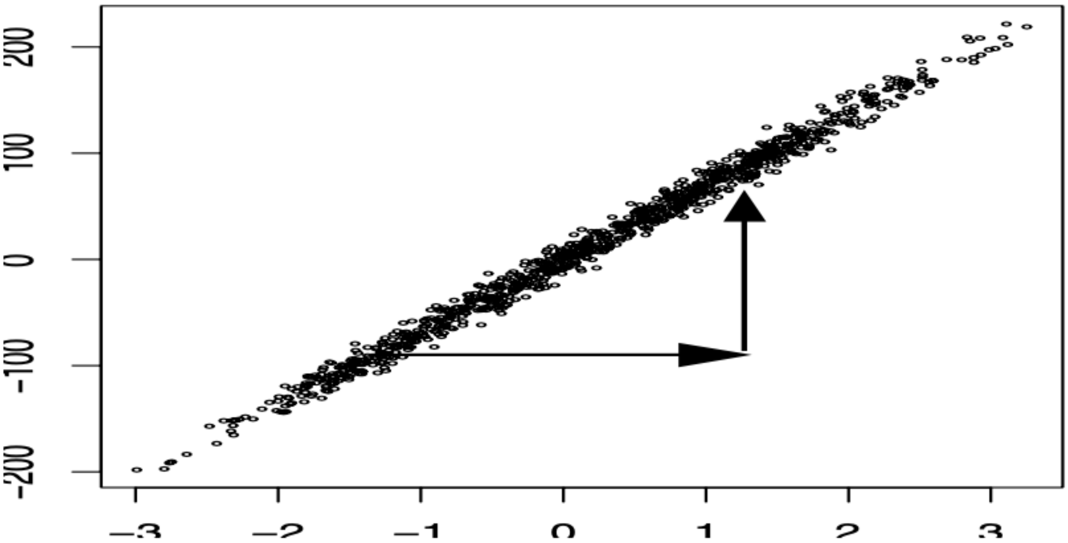


Figure A.1: The figure taken from [10, Figure 4.1 (b)], shows samples from a marginal chain for μ and $\mathbf{1}^T \boldsymbol{Q}\boldsymbol{x}^{(k)}$ over 1000 iterations, based on the hierarchical model in Eq. A.1, with an autoregressive process encoded in $\boldsymbol{Q}$. The algorithm updates μ and $\boldsymbol{x}$ successively from their full conditional distributions. The plot displays $(\mu^{(k)}, \mathbf{1}^T \boldsymbol{Q}\boldsymbol{x}^{(k)})$, with $\mu^{(k)}$ on the horizontal axis and $\mathbf{1}^T \boldsymbol{Q}\boldsymbol{x}^{(k)}$ on the vertical axis. The slow mixing and convergence of μ result from its strong dependence on $\mathbf{1}^T \boldsymbol{Q}\boldsymbol{x}^{(k)}$, while the sampler permits only axis-aligned (horizontal and vertical) and does not allow diagonal moves, as illustrated by the arrows.

B
Mesure theroy

Recall the probability space $(\Omega, \mathcal{F}, \mathbb{P})$, where Ω denotes the sample space, and $\mathcal{F}$ is a collection of countable subsets $\{A_n\}_{n\in\mathbb{N}}$ of Ω. Each $A_n \subseteq \Omega$ is called an event, and a map $\mathbb{P} : \mathcal{F} \to \mathbb{R}$ is referred to as a measure. In the following, we describe the conditions required for $\mathcal{F}$ to be a σ-algebra, and for $\mathbb{P}$ to qualify as a probability measure. We refer to [28] [20] for further reading.

B.1 probailty measure

For a probability measure, we require:

- $\mathbb{P}(\Omega) = 1$ and $\mathbb{P}(\emptyset) = 0$
- $\mathbb{P}(A) \in [0, 1]$
- $\mathbb{P}(\bigcup_{j\in\mathbb{N}} A_j) = \sum_{j\in\mathbb{N}} \mathbb{P}(A_j)$ if we have pairwise disjoint sets or $A_i \cap A_j = \emptyset$ for $i \neq j$

In other words, the probability assigned to the entire sample space must be equal to one, $\mathbb{P}(\Omega) = 1$, and the probability of the empty set must be zero, $\mathbb{P}(\emptyset) = 0$. For any subset $A \subseteq \Omega$, the probability $\mathbb{P}(A)$ must lie between zero and one, i.e., $\mathbb{P}(A) \in [0, 1]$. If e.g. two subsets A and B are disjoint (i.e., $A \cap B = \emptyset$), then the probability of their union satisfies $\mathbb{P}(A \cup B) = \mathbb{P}(A) + \mathbb{P}(B)$. This property must also hold for a countable sequence of disjoint sets $\{A_j\}_{j\in\mathbb{N}}$, such that $\mathbb{P}\left(\bigcup_{j\in\mathbb{N}} A_j\right) = \sum_{j\in\mathbb{N}} \mathbb{P}(A_j)$.

B.2 σ-algebra

A collections of subsets $\mathcal{F}$ is called σ-algebra if:

- $\emptyset, \Omega \in \mathcal{F}$,
- if $A \in \mathcal{F}$ then $A^C := A/\Omega \in \mathcal{F}$
- if $A_1, A_2, \cdots \in \mathcal{F}$ then $\bigcup_{j\in\mathbb{N}} A_j \in \mathcal{F}$

In other words, the empty set $\emptyset$ and the entire sample space Ω must always be elements of $\mathcal{F}$. If a set $A \in \mathcal{F}$, then its complement $A^C = \Omega \setminus A$ must also be in $\mathcal{F}$. If, in terms of a probability measure, we are able to assign a probability $\mathbb{P}(A)$ to an event A, we must also be able to assign a probability to the event “not A”, i.e., $\mathbb{P}(A^C)$. Finally, if a countable collection of sets $A_1, A_2, \cdots \in \mathcal{F}$, then their union $\bigcup_{j\in\mathbb{N}} A_j$ must also be in $\mathcal{F}$. These three properties define the requirements for $\mathcal{F}$ to be a σ-algebra.

References

[1] C. Readings. *Envisat MIPAS An Instrument for Atmospheric Chemistry and Climate Research.* Noordwijk: ESA Publications Division, 2000.

[2] Iouli E Gordon et al. “The HITRAN2020 molecular spectroscopic database”. In: *Journal of Quantitative Spectroscopy and Radiative Transfer* 277 (2022), p. 107949.

[3] George B. Rybicki and Alan P. Lightman. *Radiative processes in Astrophysics.* Weinheim: Wiley-VCH, 2004.

[4] Marcel Berger. *Geometry I. 4th Edition.* Berlin Heidelberg: Springer-Verlag, 2009.

[5] Katsumi Nomizu and Takeshi Sasaki. *Affine differential geometry.* Cambridge: Cambridge University Press, 1994.

[6] Colin Fox and Richard A Norton. “Fast sampling in a linear-Gaussian inverse problem”. In: *SIAM/ASA Journal on Uncertainty Quantification* 4.1 (2016), pp. 1191–1218.

[7] S. P. Meyn and R.L. Tweedie. *Markov Chains and Stochastic Stability. 2nd Edition.* New York: Cambridge University Press, 2009.

[8] Gareth O. Roberts and Jeffrey S Rosenthal. “General state space Markov chains and MCMC algorithms”. In: *Probability Surveys* 1 (2004), pp. 20–71.

[9] Charles J Geyer. “Practical markov chain monte carlo”. In: *Statistical science* (1992), pp. 473–483.

[10] Havard Rue and Leonhard Held. *Gaussian Markov random fields: theory and applications.* London: CRC press, 2005.

[11] Charles W. Champ and Andrew V. Sills. “The Generalized Law of Total Covariance”. In: *preprint* (2022). URL: `https://arxiv.org/abs/2205.14525`.

[12] Jari P. Kaipio and Erkki Somersalo. *Statistical and Computational Inverse Problems.* New York: Springer-Verlag New York, 2005.

[13] Sze M Tan, Colin Fox, and Geoff K. Nicholls. *Course notes for ELEC 445 – Inverse Problems and Imaging.* `https://coursesupport.physics.otago.ac.nz/wiki/pmwiki.php/ELEC445/HomePage`. [Online; accessed 10/12/23]. 2016.

[14] Per Christian Hansen and Dianne Prost O’Leary. “The use of the L-curve in the regularization of discrete ill-posed problems”. In: *SIAM Journal on Scientific Computing* 14.6 (1993), pp. 1487–1503.

[15] Gareth O. Roberts and Jeffrey S Rosenthal. “Harris recurrence of Metropolis-within-Gibbs and trans-dimensional Markov chains”. In: *The Annals of Applied Probability* 16.4 (2006), 2123–2139.

[16] Johnathan M Bardsley. “MCMC-based image reconstruction with uncertainty quantification”. In: *SIAM Journal on Scientific Computing* 34.3 (2012), A1316–A1332.

[17] Johnathan M Bardsley et al. “Randomize-then-optimize: A method for sampling from posterior distributions in nonlinear inverse problems”. In: *SIAM Journal on Scientific Computing* 36.4 (2014), A1895–A1910.

[18] Felipe Acosta, Mark L Huber, and Galin L Jones. “Markov chain Monte Carlo with linchpin variables”. In: *preprint* (2014). URL: `https://arxiv.org/abs/2205.14525`.

[19] J. Andrés Christen and Colin Fox. "A general purpose sampling algorithm for continuous distributions (the t-walk)". In: *Bayesian Analysis* 5.2 (2010), pp. 263 –281. URL: `https://doi.org/10.1214/10-BA603`.

[20] M. Capiński and P.E. Kopp. *Measure, Integral and Probability. Springer Undergraduate Mathematics Series.* London: Springer-Verlag London, 2004.

[21] M. Simonnet. *Measures and Probabilities.* New York: Springer-Verlag, 1996.

[22] Vesa Kaarnioja. *Inverse Problems. Eighth lecture.* `https://vesak90.userpage.fu-berlin.de/ip23/week8.pdf`. [Online; accessed 10/04/25]. 2023.

[23] Tiangang Cui and Sergey Dolgov. "Deep composition of tensor-trains using squared inverse rosenblatt transports". In: *Foundations of Computational Mathematics* 22.6 (2022), pp. 1863–1922.

[24] Philip J Davis and Philip Rabinowitz. *Methods of numerical integration.* San Diego, CA: Academic Press, Inc., 1984.

[25] Colin Fox et al. "Grid methods for Bayes-optimal continuous-discrete filtering and utilizing a functional tensor train representation". In: *Inverse Problems in Science and Engineering* 29.8 (2021), pp. 1199–1217.

[26] Sergey Dolgov et al. "Approximation and sampling of multivariate probability distributions in the tensor train decomposition". In: *Statistics and Computing* 30 (2020), pp. 603–625.

[27] Ivan V Oseledets. "Tensor-train decomposition". In: *SIAM Journal on Scientific Computing* 33.5 (2011), pp. 2295–2317.

[28] Greg Lawler. *Notes on probability.* `https://www.math.uchicago.edu/~lawler/probnotes.pdf`. [Online; accessed 10/04/25]. 2016.