Contents

List of Figures iii

1 Introduction 3
- 1.1 Motivation . . . 3
- 1.2 What is going on?, 3 facts, What is new in this thesis? . . . 3
- 1.3 Thesis Outline . . . 3

2 Theoretical and Technical Background 5
- 2.1 Affine Map . . . 5
- 2.2 Bayesian Inference . . . 6
  - 2.2.1 Marginal and conditional posterior distribution . . . 9
- 2.3 Regularisation . . . 10
- 2.4 Sampling Methods . . . 11
  - 2.4.1 Metropolis- within Gibbs sampling . . . 11
  - 2.4.2 Draw a sample from a multivariate normal distribution . . . 12
  - 2.4.3 t-walk sampler as black box . . . 13
- 2.5 Numerical Approxiamtion Methods - Tensor Train . . . 13
  - 2.5.1 Marginal Functions . . . 16

3 Forward Model 19

4 Results and Conclusions 23
- 4.1 Simulate Data and ground truth . . . 23
- 4.2 Set up the Bayesian framework . . . 25
  - 4.2.1 Prior Modelling . . . 25
- 4.3 Posterior distributions with Linear model for Ozone – MTC . . . 32
  - 4.3.1 Hyper-parameters samples from the marginal posterior distribution 33
  - 4.3.2 Ozone samples from the conditional posterior . . . 35
- 4.4 Approximate non-linear forward model with affine Map . . . 37
- 4.5 Posterior distributions with approximated non-linear model for Ozone – MTC . . . 40
  - 4.5.1 Hyper-parameters samples from the marginal posterior distribution 40
  - 4.5.2 Ozone samples from the conditional posterior and regularised solution 42

4.5.3 Solution by regularisation 42
4.6 Posterior pressure and temperature 43
4.7 Error analysis 50

5 Summary and Outlook **53**
5.1 Atmospheric Physics 53
5.2 Methods 53

Appendices

A Additional MCMC analysis **57**

B Correlation Structure **59**

C Mesure theroy **61**
C.1 probailty measure 61
C.2 σ-algebra 62

D prior modelling **63**
D.1 t-walk 63

References **71**

List of Figures

2.1 Schematics of the affine map 6
2.2 Bayesian Inference DAG 7
2.3 Visualisation of a tensor train 15

3.1 Schematic of measurement and analysis geometry. 19

4.1 Complete directed acyclic graph of the forward model. 26
4.2 Samples from ozone prior distribution. 28
4.3 Prior Samples of $\boldsymbol{p}/\boldsymbol{T}$ according to the respective hyper-prior distribution. 29
4.4 Prior Samples of $\boldsymbol{T}$ according to the respective hyper-prior distribution. . 30
4.5 Prior Samples of $\boldsymbol{p}$ according to the respective hyper-prior distribution. . 31
4.6 Directed acyclic graph for ozone retrieval and MTC scheme. 32
4.7 Plot of the functions $f(\lambda)$ and $g(\lambda)$ for marginal posterior. 34
4.8 Scatter plot of samples from marginal posterior, including weighting from TT approximation; additional trace plot of the marginal posterior samples. 36
4.9 Ozone samples of the conditional posterior. 38
4.10 Strategy to find affine map. 38
4.11 Assessment of affine map. 40
4.12 Marginal posterior histograms and TT approximation as well as hyper-prior distribution. 41
4.13 Ozone posterior mean and variance and the regularised solution compared to the ground truth. 42
4.14 Plot of the L-curve to find the regularised solution. 44
4.15 Directed acyclic Graph for pressure and temperature. 45
4.16 Histograms and TT approximation of posterior distribution as well as hyper-prior distribution. 45
4.17 Histograms and TT approximation of posterior distribution as well as hyper-prior distribution. 46
4.18 Histograms and TT approximation of posterior distribution as well as hyper-prior distribution. 47
4.19 Histograms and TT approximation of posterior distribution as well as hyper-prior distribution. 48

4.20 Histograms and TT approximation of posterior distribution as well as hyper-prior distribution. 49
4.21 Temperature posterior samples. 50
4.22 Pressure posterior samples. 51

B.1 Correlation structure in between parameters and hyper-parameters 60

421.10046pt

1 Introduction

1.1 Motivation

- ozone coverage
- regularisation approach in atmospheric physics citation
- 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

Note the following: In this case the best fit to data is not the best fit to parameters. In under- graduate statistics courses you would learn the more general notion that: "Conditioning on estimates gives poor predictive densities". [1]

421.10046pt

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 4. We begin by introducing the forward model in Section **??**, which we use to simulate the data. Since the forward model is weakly non-linear, we employ an affine transformation, see Section 2.1, 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 2.2, 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 2.3. In Section 2.4, we introduce Markov Chain Monte Carlo (MCMC) methods to sample from the posterior distribution Finally, in Section 2.5, instead of sampling, we can approximate the posterior distribution using the tensor train (TT) format.

2.1 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 [2, 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 [2, 3]

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

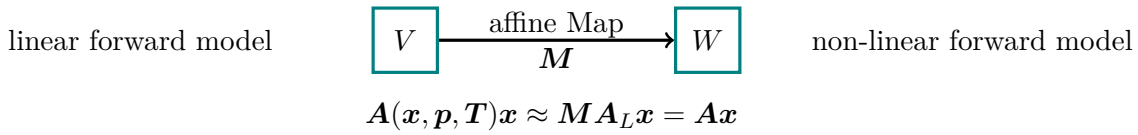

Figure 2.1: This Figure shows the schematic representation of the affine map $\boldsymbol{M}$, which approximates the non-linear forward model from the linear forward model. Here, V contains values produced by the linear forward model, and W contains the corresponding values from the non-linear forward 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.

we generate two affine subspaces spaces $V = \{\boldsymbol{A}(\boldsymbol{x}^{(1)}, \boldsymbol{p}, \boldsymbol{T}), \ldots, \boldsymbol{A}(\boldsymbol{x}^{(m)}, \boldsymbol{p}, \boldsymbol{T})\}$ and $W = \{\boldsymbol{A}_L\boldsymbol{x}^{(1)}, \ldots, \boldsymbol{A}_L\boldsymbol{x}^{(m)}\}$ 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 $\{\boldsymbol{x}^{(1)}, \ldots, \boldsymbol{x}^{(m)}\} \sim \pi(\boldsymbol{x}|\boldsymbol{\theta}, \boldsymbol{y})$, 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.

2.2 Bayesian Inference

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

$$\boldsymbol{y} = \boldsymbol{A}\boldsymbol{x} + \boldsymbol{\eta}, \tag{2.1}$$

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 **??**.

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 2.2. Since any observational process naturally involves random noise, we include this in the DAG and classify the noise variance as a hyper-parameter within $\boldsymbol{\theta}$ [4]. 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 2.2, 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 physical properties or functional dependences. This is one of the great strength of Bayesian modelling compared to e.g. regularisation, see section 2.3. 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 2.2.

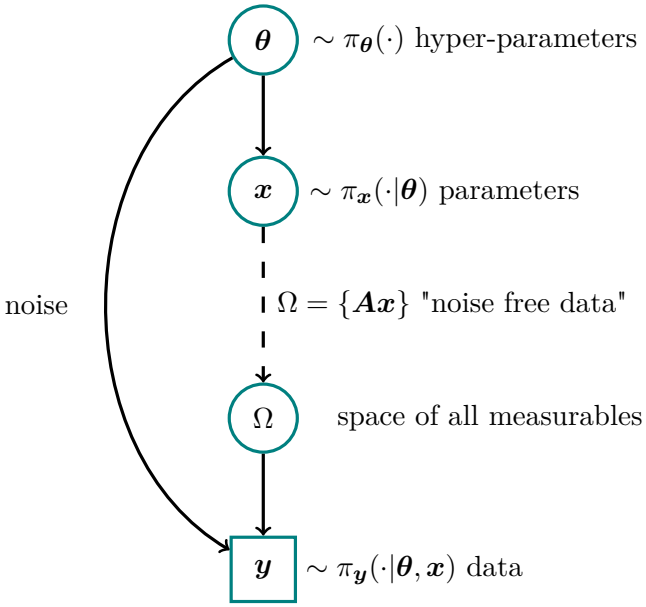

Figure 2.2: The directed acyclic graph (DAG) for a linear inverse problem visualises statistical dependencies as solid line arrows and deterministic dependencies as dotted arrows. The parameters $\boldsymbol{x}$ have some statistical dependency of those hyper-parameters $\boldsymbol{\theta}$, which are 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 $\boldsymbol{u} = \boldsymbol{A}\boldsymbol{x}$ deterministically through the linear forward model $\boldsymbol{A}$. From the space of all measurables we observe some data $\boldsymbol{y} = \boldsymbol{A}\boldsymbol{x} + \boldsymbol{\eta}$, statistically, so that $\boldsymbol{y} \sim \pi_{\boldsymbol{y}}(\cdot|\boldsymbol{\theta},\boldsymbol{x})$, with naturally some random noise $\boldsymbol{\eta} \sim \pi_{\boldsymbol{\eta}}(\cdot|\boldsymbol{\theta})$.

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

$$\pi(\boldsymbol{x},\boldsymbol{\theta}|\boldsymbol{y}) = \frac{\pi(\boldsymbol{y}|\boldsymbol{x},\boldsymbol{\theta})\pi(\boldsymbol{x},\boldsymbol{\theta})}{\pi(\boldsymbol{y})}\,, \tag{2.2}$$

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

$$\pi(\boldsymbol{x},\boldsymbol{\theta}|\boldsymbol{y}) \propto \pi(\boldsymbol{y}|\boldsymbol{x},\boldsymbol{\theta})\pi(\boldsymbol{x},\boldsymbol{\theta})\,. \tag{2.3}$$

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

$$\mathrm{E}_{\boldsymbol{x},\boldsymbol{\theta}|\boldsymbol{y}}[h(\boldsymbol{x}_{\boldsymbol{\theta}})] = \underbrace{\int\int h(\boldsymbol{x}_{\boldsymbol{\theta}})\,\pi(\boldsymbol{x},\boldsymbol{\theta}|\boldsymbol{y})\,\mathrm{d}\boldsymbol{x}\,\mathrm{d}\boldsymbol{\theta}}_{\boldsymbol{\mu}_{\mathrm{int}}}\,, \tag{2.4}$$

which may be a high dimensional integral and computationally not feasible to solve. Therefore the unbiased [5] sample based Monte Carlo estimate

$$\mathrm{E}_{\boldsymbol{x},\boldsymbol{\theta}|\boldsymbol{y}}[h(\boldsymbol{x}_{\boldsymbol{\theta}})] \approx \underbrace{\frac{1}{N}\sum_{k=1}^{N} h(\boldsymbol{x}_{\boldsymbol{\theta}}^{(k)})}_{\boldsymbol{\mu}_{\mathrm{samp}}}, \tag{2.5}$$

for large enough N (law of large numbers [6, Chapter 17]) is often used. Here, the samples $\{\boldsymbol{x}^{(k)}, \boldsymbol{\theta}^{(k)}\} \sim \pi_{\boldsymbol{x},\boldsymbol{\theta}}(\cdot|\boldsymbol{y})$, for $k = 1, \ldots, N$, form a sample set $\mathcal{M} = \{(\boldsymbol{x}, \boldsymbol{\theta})^{(1)}, \ldots, (\boldsymbol{x}, \boldsymbol{\theta})^{(N)}\}$. Furthermore, the central limit theorem states that the samples mean $\boldsymbol{\mu}_{\mathrm{samp}}^{(i)}$, of independent samples sets $\mathcal{M}_i$ for $i = 1, \ldots, n$ of any distribution, converge in distribution to a normal distribution so that

$$\sqrt{n}(\boldsymbol{\mu}_{\mathrm{samp}}^{(i)} - \boldsymbol{\mu}_{\mathrm{int}}) \xrightarrow{\mathcal{D}} \mathcal{N}(0, \sigma^2)[7], \tag{2.6}$$

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

On the Monte-Carlo Error and Integrated Autocorrelation time

The error is for one markov chain $\mathcal{M}_i$

$$\sigma^2 = \mathrm{var}(\boldsymbol{\mu}_{\mathrm{samp}}^{(i)}) = \mathrm{var}(\mathrm{E}_{\boldsymbol{x},\boldsymbol{\theta}|\boldsymbol{y}}[h(\boldsymbol{x}_{\boldsymbol{\theta}})]) = \left(\frac{1}{N}\sum_{k=1}^{N} h(\boldsymbol{x}_{\boldsymbol{\theta}}^{(k)}) - \boldsymbol{\mu}^{(i)}\right)^2 \tag{2.7}$$

$$\sigma^2 = \mathrm{var}(\boldsymbol{\mu}_{\mathrm{samp}}^{(i)}) = \mathrm{var}(\mathrm{E}_{\boldsymbol{x},\boldsymbol{\theta}|\boldsymbol{y}}[h(\boldsymbol{x}_{\boldsymbol{\theta}})]) = \frac{1}{N^2}\sum_{k=1}^{N} \rho_{k,s} \tag{2.8}$$

We need to define a correlation function why?

$$C(t) \equiv \langle x_s x_{s+t} - \mu \rangle \tag{2.9}$$

where $C(0) = \mathrm{var}(x)$ The auto covariance function or auto correlation function is given as

The integrated autocorrelation time (IACT)

$$\tau_{\mathrm{int}} = 1 + 2\sum_{k=1}^{\infty} \rho_k, \tag{2.10}$$

as defined in [8] provides a good estimate on how efficient a sampler is, where ρ_k is the normalised autocorrelation at lag k. More specifically, the IACT gives an estimate of how many steps the sampling algorithm needs to take to produce one independent sample. We calculate the IACT using the pyhton implementation of [9] and double the output provided.

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 B. 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

$$\pi(\boldsymbol{x}, \boldsymbol{\theta}|\boldsymbol{y}) = \pi(\boldsymbol{x}|\boldsymbol{\theta}, \boldsymbol{y})\, \pi(\boldsymbol{\theta}|\boldsymbol{y}), \tag{2.11}$$

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. (2.4) becomes

$$\mathbb{E}_{\boldsymbol{x}|\boldsymbol{y}}[h(\boldsymbol{x})] = \mathbb{E}_{\boldsymbol{\theta}|\boldsymbol{y}}\left[\mathbb{E}_{\boldsymbol{x}|\boldsymbol{\theta},\boldsymbol{y}}[h(\boldsymbol{x}_{\boldsymbol{\theta}})]\right] = \int \mathbb{E}_{\boldsymbol{x}|\boldsymbol{\theta},\boldsymbol{y}}\left[h(\boldsymbol{x}_{\boldsymbol{\theta}})\right]\, \pi(\boldsymbol{\theta}|\boldsymbol{y})\, \mathrm{d}\boldsymbol{\theta}, \tag{2.12}$$

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 [4]

$$\boldsymbol{y}|\boldsymbol{x}, \boldsymbol{\theta} \sim \mathcal{N}(\boldsymbol{A}\boldsymbol{x}, \boldsymbol{\Sigma}(\boldsymbol{\theta})) \tag{2.13a}$$

$$\boldsymbol{x}|\boldsymbol{\theta} \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{Q}^{-1}(\boldsymbol{\theta})) \tag{2.13b}$$

$$\boldsymbol{\theta} \sim \pi(\boldsymbol{\theta}), \tag{2.13c}$$

with a normally distributed likelihood $\pi(\boldsymbol{y}|\boldsymbol{x}, \boldsymbol{\theta})$ Why? other likelihoods? 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.

2.2.1 Marginal and conditional posterior distribution

For the linear-Gaussian Bayesian hierarchical model specified in Eq. 2.13, the marginal posterior distribution over the hyper-parameters is given by

$$\pi(\boldsymbol{\theta}|\boldsymbol{y}) = \int \pi(\boldsymbol{x}, \boldsymbol{\theta}|\boldsymbol{y})\, \mathrm{d}\boldsymbol{x} \tag{2.14}$$

$$\propto \sqrt{\frac{\det(\boldsymbol{\Sigma}^{-1})\, \det(\boldsymbol{Q})}{\det(\boldsymbol{Q} + \boldsymbol{A}^T\boldsymbol{\Sigma}^{-1}\boldsymbol{A})}} \times \exp\left[-\frac{1}{2}(\boldsymbol{y} - \boldsymbol{A}\boldsymbol{\mu})^T \boldsymbol{Q}_{\boldsymbol{\theta}|\boldsymbol{y}} (\boldsymbol{y} - \boldsymbol{A}\boldsymbol{\mu})\right] \pi(\boldsymbol{\theta})\,, \tag{2.15}$$

with

$$\boldsymbol{Q}_{\boldsymbol{\theta}|\boldsymbol{y}} = \boldsymbol{\Sigma}^{-1} - \boldsymbol{\Sigma}^{-1}\boldsymbol{A}\left(\boldsymbol{A}^T\boldsymbol{\Sigma}^{-1}\boldsymbol{A} + \boldsymbol{Q}\right)^{-1}\boldsymbol{A}^T\boldsymbol{\Sigma}^{-1}\,, \tag{2.16}$$

see [4, Lemma 2]. Conditioned on the hyper-parameters $\boldsymbol{\theta}$, we can draw samples from the normal conditional posterior distribution

$$\boldsymbol{x}|\boldsymbol{\theta},\boldsymbol{y} \sim \mathcal{N}\Big(\underbrace{\boldsymbol{\mu} + \Big(\boldsymbol{A}^T\boldsymbol{\Sigma}^{-1}\boldsymbol{A} + \boldsymbol{Q}\Big)^{-1}\boldsymbol{A}^T\boldsymbol{\Sigma}^{-1}(\boldsymbol{y} - \boldsymbol{A}\boldsymbol{\mu})}_{\boldsymbol{\mu}_{\boldsymbol{x}|\boldsymbol{\theta},\boldsymbol{y}}}, \underbrace{\Big(\boldsymbol{A}^T\boldsymbol{\Sigma}^{-1}\boldsymbol{A} + \boldsymbol{Q}\Big)^{-1}}_{\boldsymbol{\Sigma}_{\boldsymbol{x}|\boldsymbol{y},\boldsymbol{\theta}}}\Big), \quad (2.17)$$

using the Randomise-then-Optimise (RTO) method (see Section 2.4.2), or compute weighted expectations, as in Eq. 2.12, 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 $\boldsymbol{\theta}$.

2.3 Regularisation

Another method for obtaining a solution to the linear inverse problem in Eq. 2.1 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 [4]. Here we focus on a regularisation semi-norm for the case of Tikhonov regularisation [1, 12], which is closest to a linear-Gaussian hierarchical Bayesian model, as introduced in Eq. 2.13.

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

$$\|\boldsymbol{y} - \boldsymbol{A}\boldsymbol{x}\| \quad (2.18)$$

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

$$\lambda \|\boldsymbol{T}\boldsymbol{x}\| \quad (2.19)$$

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

$$\boldsymbol{x}_\lambda = \arg\min_{\boldsymbol{x}} \|\boldsymbol{y} - \boldsymbol{A}\boldsymbol{x}\|^2 + \lambda \|\boldsymbol{T}\boldsymbol{x}\|^2 \quad (2.20)$$

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

$$\nabla_{\boldsymbol{x}} \Big\{(\boldsymbol{y} - \boldsymbol{A}\boldsymbol{x})^T(\boldsymbol{y} - \boldsymbol{A}\boldsymbol{x}) + \lambda \boldsymbol{x}^T\boldsymbol{T}^T\boldsymbol{T}\boldsymbol{x}\Big\} = 0 \quad (2.21)$$

$$\iff \quad \nabla_{\boldsymbol{x}} \Big\{\boldsymbol{y}^T\boldsymbol{y} + \boldsymbol{x}^T\boldsymbol{A}^T\boldsymbol{A}\boldsymbol{x} - 2\boldsymbol{y}^T\boldsymbol{A}\boldsymbol{x} + \lambda \boldsymbol{x}^T\boldsymbol{T}^T\boldsymbol{T}\boldsymbol{x}\Big\} = 0 \quad (2.22)$$

$$\iff \quad 2\boldsymbol{A}^T\boldsymbol{A}\boldsymbol{x} - 2\boldsymbol{A}^T\boldsymbol{y} + 2\lambda\boldsymbol{T}^T\boldsymbol{T}\boldsymbol{x} = 0. \quad (2.23)$$

Solving this equation yields the regularised solution

$$\boldsymbol{x}_\lambda = (\boldsymbol{A}^T\boldsymbol{A} + \lambda\boldsymbol{L})^{-1}\boldsymbol{A}^T\boldsymbol{y}\,, \quad (2.24)$$

where we define $\boldsymbol{L} := \boldsymbol{T}^T\boldsymbol{T}$, which typically represents a discrete approximation of a derivative operator [1].

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 [13], which we plot in Section **??**.

2.4 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. 2.5. 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 [1, 5].

The ergodicity theorem in [1] 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 [1]), 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.

2.4.1 Metropolis- within Gibbs sampling

As introduced in Section 2.2.1, 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 [4]. 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 [14].

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 [14]. We accept and set $\theta_1^{(t)} = \theta_1$ with the acceptance probability

$$\alpha(\theta_1|\theta_1^{(t-1)}) = \min\left\{1, \frac{\pi(\theta_1|\theta_2^{(t-1)}, \boldsymbol{y})\, \cancel{q(\theta_1^{(t-1)}|\theta_1)}}{\pi(\theta_1^{(t-1)}|\theta_2^{(t-1)}, \boldsymbol{y})\, \cancel{q(\theta_1|\theta_1^{(t-1)})}}\right\} \tag{2.25}$$

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

Initialise and suppose two dimensional vector $\boldsymbol{\theta}^{(0)} = (\theta_1^{(0)}, \theta_2^{(0)})$
for $k = 1, \ldots, N'$ **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: $\boldsymbol{\theta}^{(0)}, \ldots, \boldsymbol{\theta}^{(k)}, \ldots, \boldsymbol{\theta}^{(N)} \sim \pi(\boldsymbol{\theta}|\boldsymbol{y})$

2.4.2 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 [15].

The full conditional distribution can be rewritten as

$$\pi(\boldsymbol{x}|\boldsymbol{y}, \boldsymbol{\theta}) \propto \pi(\boldsymbol{y}|\boldsymbol{x}, \boldsymbol{\theta})\, \pi(\boldsymbol{x}|\boldsymbol{\theta}) \tag{2.26}$$

$$= \exp\left(-\left\|\hat{\boldsymbol{A}}\boldsymbol{x} - \hat{\boldsymbol{y}}\right\|^2\right), \tag{2.27}$$

where

$$\hat{\boldsymbol{A}} = \begin{bmatrix} \boldsymbol{\Sigma}^{-1/2}(\boldsymbol{\theta})\boldsymbol{A} \\ \boldsymbol{Q}^{1/2}(\boldsymbol{\theta}) \end{bmatrix}, \quad \hat{\boldsymbol{y}} = \begin{bmatrix} \boldsymbol{\Sigma}^{-1/2}(\boldsymbol{\theta})\boldsymbol{y} \\ \boldsymbol{Q}^{1/2}(\boldsymbol{\theta})\boldsymbol{\mu} \end{bmatrix} \quad [16]. \tag{2.28}$$

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

$$\boldsymbol{x}_i = \arg\min_{\hat{\boldsymbol{x}}} \|\hat{\boldsymbol{A}}\hat{\boldsymbol{x}} - (\hat{\boldsymbol{y}} + \boldsymbol{b})\|^2, \quad \boldsymbol{b} \sim \mathcal{N}(\boldsymbol{0}, \mathbf{I}), \tag{2.29}$$

where we add a randomised perturbation $\boldsymbol{b}$. Similar to Section 2.3, this expression can be rewritten as

$$\left(\boldsymbol{A}^T\boldsymbol{\Sigma}^{-1}(\boldsymbol{\theta})\boldsymbol{A} + \boldsymbol{Q}(\boldsymbol{\theta})\right)\boldsymbol{x}_i = \boldsymbol{A}^T\boldsymbol{\Sigma}^{-1}(\boldsymbol{\theta})\boldsymbol{y} + \boldsymbol{Q}(\boldsymbol{\theta})\boldsymbol{\mu} + \boldsymbol{v}_1 + \boldsymbol{v}_2, \tag{2.30}$$

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 [4, 15].

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 $\{(\boldsymbol{x}, \boldsymbol{\theta})^{(1)}, \ldots, (\boldsymbol{x}, \boldsymbol{\theta})^{(N)}\} \sim \pi(\boldsymbol{x}, \boldsymbol{\theta}|\boldsymbol{y})$ is also ergodic [17].

2.4.3 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 [18]. 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.

2.5 Numerical Approxiamtion Methods - Tensor Train

Explain how to find normalisation constant and say that due to aproxiamting the sqaure root we ensure posivyt later when squaring it 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. The motivation to use the tensor train format is that we can approximate a d-dimensional grid with far fewer data points compared to the total number of grid points.

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 C. We denote

$$\mathbb{P}(A) = \int_A \mathrm{d}\mathbb{P} \tag{2.31}$$

as the probability of an event $A \in \mathcal{F}$. By applying the Radon-Nikodym theorem [19], we can change variables

$$\mathbb{P}(A) = \int_A \frac{\mathrm{d}\mathbb{P}}{\mathrm{d}x}\,\mathrm{d}x = \int_A \pi(x)\,\mathrm{d}x, \tag{2.32}$$

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 [20, 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 [19, 21]. As by Cui et al. [22], 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

$$f_{X_k}(x_k) = \int_{\mathcal{X}_1} \cdots \int_{\mathcal{X}_d} \boldsymbol{\lambda}(x)\,\pi(x)\,\mathrm{d}x_1 \cdots \mathrm{d}x_{k-1}\,\mathrm{d}x_{k+1} \cdots \mathrm{d}x_d, \tag{2.33}$$

where we integrate over all dimensions except the k-th. Here, we introduce a weight function $\boldsymbol{\lambda}(x)$ [23], which can be useful for quadrature rules??. Cui et al. [22] refer to $\boldsymbol{\lambda}(x)$ as a "product-form Lebesgue-measurable weighting function" and define it as

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

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 2D matrices (rank-2 tensor) and 3D matrices (rank-3 tensors), which we call TT-cores, see Fig. 2.3. 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 2.3. 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

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

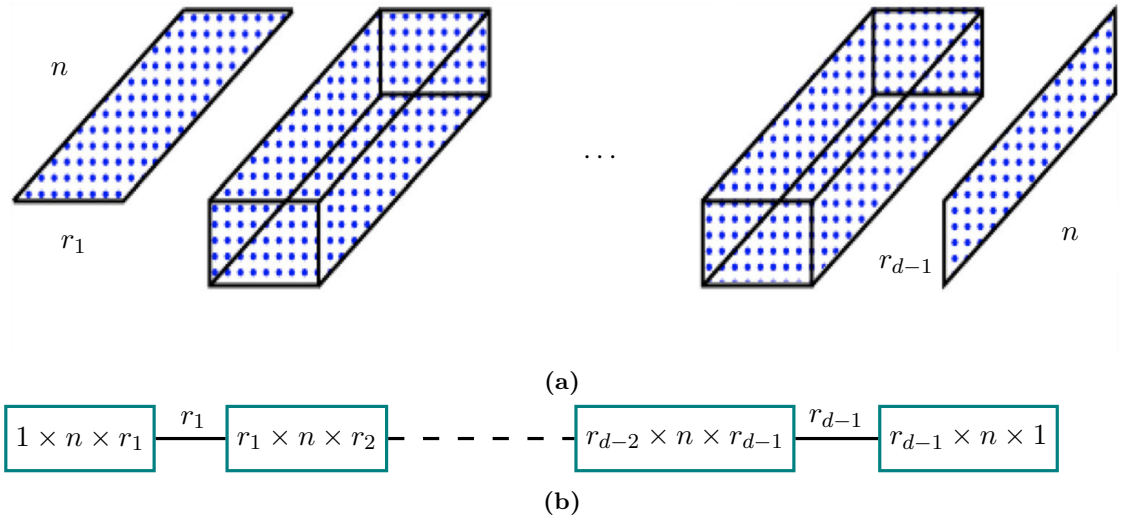

Figure 2.3: Here, we visualise the tensor train cores as two- and three-dimensional matrices. Each core has a length n, corresponding to the number of grid points in one dimension, and the cores are connected through ranks r_k. More specifically, a core π_k has dimensions $r_{k-1} \times n \times r_k$, with outer ranks $r_0 = r_d = 1$. Using the TT-format enables us to represent a d-dimensional grid with only dnr^2 evaluation points instead of n^d grid points. Figure (a) is adapted from [24].

where each core $\pi_k(x_k)$, becomes a matrix of size $r_{k-1} \times r_k$. Clearly this shows that we only need dnr^2 evaluation points instead of n^d grid points to approximate the whole parameter space. Consequently, with a tensor train approximation, the marginal target function

$$f_{X_k}(x_k) = \frac{1}{z} \Big| \left(\int_{\mathbb{R}} \boldsymbol{\lambda}_1(x_1)\pi_1(x_1)\,\mathrm{d}x_1 \right) \cdots \left(\int_{\mathbb{R}} \boldsymbol{\lambda}_{k-1}(x_{k-1})\pi_{k-1}(x_{k-1})\,\mathrm{d}x_{k-1} \right) \boldsymbol{\lambda}_k(x_k)\pi_k(x_k) \left(\int_{\mathbb{R}} \boldsymbol{\lambda}_{k+1}(x_{k+1})\pi_{k+1}(x_{k+1})\,\mathrm{d}x_{k+1} \right) \cdots \left(\int_{\mathbb{R}} \boldsymbol{\lambda}_d(x_d)\pi_d(x_d)\,\mathrm{d}x_d \right) \Big| \tag{2.34}$$

is computed by integrating over all TT cores except π_k, as in [25], including a normalisation constant z [22].

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. [22] and instead approximate the square root of the probability density

$$\sqrt{\pi(x)} \approx g(x) = \boldsymbol{G}_1(x_1), \ldots, \boldsymbol{G}_k(x_k), \ldots, \boldsymbol{G}_d(x_d). \tag{2.35}$$

Here, each TT-core is given by

$$G_k^{(\alpha_{k-1},\alpha_k)}(x_k) = \sum_{i=1}^{n_k} \phi_k^{(i)}(x_k)\boldsymbol{A}_k[\alpha_{k-1}, i, \alpha_k], \quad k = 1, \ldots, d, \tag{2.36}$$

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:

$$\pi(x) \approx \gamma' + g^2(x), \tag{2.37}$$

where γ' is a positive constant added according to the absolute error and the Lebesgue weighting, see Eq. 2.34, to ensure positivity such that

$$\gamma' \leq \frac{1}{\lambda(\mathcal{X})} \|g - \sqrt{\pi}\|_2^2. \tag{2.38}$$

This leads to the normalised target function

$$f_X(x) = \frac{1}{z}\lambda(x)\pi(x) = \frac{1}{z}\left(\lambda(x)\gamma' + \lambda(x)g^2(x)\right), \tag{2.39}$$

where z is the normalisation constant, which calculate numerically within the process of finding the marginals Given the tensor train approximation of $\sqrt{\pi}$, the marginal function $f_{X_k}(x_k)$ can be expressed as

$$\begin{aligned} f_{X_k}(x_k) = \frac{1}{z}\Bigg(&\gamma' \prod_{i=1}^{k-1} \lambda_i(\mathcal{X}_i) \prod_{i=k+1}^{d} \lambda_i(\mathcal{X}_i) \\ &+ \left(\int_{\mathbb{R}} \lambda_1(x_1)\boldsymbol{G}_1^2(x_1)\,\mathrm{d}x_1\right) \cdots \left(\int_{\mathbb{R}} \lambda_{k-1}(x_{k-1})\boldsymbol{G}_{k-1}^2(x_{k-1})\,\mathrm{d}x_{k-1}\right) \\ &\lambda_k(x_k)\boldsymbol{G}_k^2(x_k) \\ &\left(\int_{\mathbb{R}} \lambda_{k+1}(x_{k+1})\boldsymbol{G}_{k+1}^2(x_{k+1})\,\mathrm{d}x_{k+1}\right) \cdots \left(\int_{\mathbb{R}} \lambda_d(x_d)\boldsymbol{G}_d^2(x_d)\,\mathrm{d}x_d\right)\Bigg). \end{aligned} \tag{2.40}$$

To compute these marginals efficiently, one can use a procedure similar to left and right orthogonalisation of TT-cores [26]. For this, we define the mass matrix $\boldsymbol{M}_k \in \mathbb{R}^{n_k \times n_k}$ as

$$\boldsymbol{M}_k[i,j] = \int_{\mathcal{X}_k} \phi_k^{(i)}(x_k)\phi_k^{(j)}(x_k)\lambda(x_k)\,\mathrm{d}x_k, \quad i,j = 1,\ldots,n_k, \tag{2.41}$$

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

2.5.1 Marginal Functions

We compute the marginal functions using two procedures, referred to as backward marginalisation [22] 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 [22]. The proposition used to compute $\boldsymbol{B}_k$, stated in Proposition 1, is adapted directly from [22].

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 $\boldsymbol{C}_k \in \mathbb{R}^{r_{k-1} \times n_k \times r_k}$:

$$\boldsymbol{C}_k[\alpha_{k-1}, \tau, l_k] = \sum_{i=1}^{n_k} \boldsymbol{B}_k[\alpha_{k-1}, i, l_k] \boldsymbol{L}_k[i, \tau]. \tag{2.42}$$

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

$$\boldsymbol{Q}_k \boldsymbol{R}_k = (\boldsymbol{C}_k^{(R)})^\top. \tag{2.43}$$

3. Compute the new coefficient tensor:

$$\boldsymbol{B}_{k-1}[\alpha_{k-2}, i, l_{k-1}] = \sum_{\alpha_{k-1}=1}^{r_{k-1}} \boldsymbol{A}_{k-1}[\alpha_{k-2}, i, \alpha_{k-1}] \boldsymbol{R}_k[l_{k-1}, \alpha_{k-1}]. \tag{2.44}$$

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 $\boldsymbol{C}_k \in \mathbb{R}^{r_{k-1} \times n_k \times r_k}$:

$$\boldsymbol{C}_{\text{pre},k}[\alpha_{k-1}, \tau, l_k] = \sum_{i=1}^{n_k} \boldsymbol{L}_k[i, \tau] \boldsymbol{B}_{\text{pre},k}[\alpha_{k-1}, i, l_k]. \tag{2.45}$$

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

$$\boldsymbol{Q}_{pre,k} \boldsymbol{R}_{\text{pre},k} = (\boldsymbol{C}_{\text{pre},k}^{(R)}). \tag{2.46}$$

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

$$\boldsymbol{B}_{\text{pre},k+1}[l_{k+1}, i, \alpha_{k+1}] = \sum_{\alpha_k=1}^{r_k} \boldsymbol{R}_{\text{pre},k}[l_{k+1}, \alpha_k] \boldsymbol{A}_{k+1}[\alpha_k, i, \alpha_{k+1}]. \tag{2.47}$$

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

$$f_{X_k}(x_k) = \frac{1}{z} \left(\gamma' \prod_{i=1}^{k-1} \lambda_i(X_i) \prod_{i=k+1}^{d} \lambda_i(X_i) + \sum_{l_{k-1}=1}^{r_{k-1}} \sum_{l_k=1}^{r_k} \left(\sum_{i=1}^{n} \phi_k^{(i)}(x_k) \boldsymbol{D}_k[l_{k-1}, i, l_k] \right)^2 \right) \lambda_k(x_k), \tag{2.48}$$

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 $\boldsymbol{B}_k \in \mathbb{R}^{r_{k-1}\times n\times r_k}$

$$\boldsymbol{D}_k[l_{k-1}, i, l_k] = \sum_{\alpha_{k-1}=1}^{r_{k-1}} \boldsymbol{R}_{\text{pre},k-1}[l_{k-1}, \alpha_{k-1}]\boldsymbol{B}_k[\alpha_{k-1}, i, l_k]. \tag{2.49}$$

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

$$f_{X_1}(x_1) = \frac{1}{z}\left(\gamma' \prod_{i=2}^{d} \boldsymbol{\lambda}_i(\mathcal{X}_i) + \sum_{l_1=1}^{r_1}\left(\sum_{i=1}^{n} \phi_1^{(i)}(x_1)\boldsymbol{D}_1[i, l_1]\right)^2\right)\boldsymbol{\lambda}_1(x_1), \tag{2.50}$$

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

$$f_{X_d}(x_d) = \frac{1}{z}\left(\gamma' \prod_{i=1}^{d-1} \boldsymbol{\lambda}_i(\mathcal{X}_i) + \sum_{l_{n-1}=1}^{r_{d-1}}\left(\sum_{i=1}^{n} \phi_1^{(i)}(x_1)\boldsymbol{D}_d[l_{n-1}, i]\right)^2\right)\boldsymbol{\lambda}_d(x_d), \tag{2.51}$$

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
Forward Model

In this chapter we present the forward model we apply all our methodology on. We follow the MIPAS handbook [27] and simulate data according to a atmosphere in local thermodynamic and and assume a measurement instrument with infinite spectral resolution and no pointing errors.

The forward model is based on a satellite measuring thermal radiation of gas molecules along its line of sight by pointing through the atmosphere to the edge(limb) of the atmosphere, known as limb sounding, as shown in Figure 3.1. One measurement y_j, of a stationary satellite is given by the path integral through the atmosphere along the line of sight. For each measurement $j = 1, 2, \ldots, m$ 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.

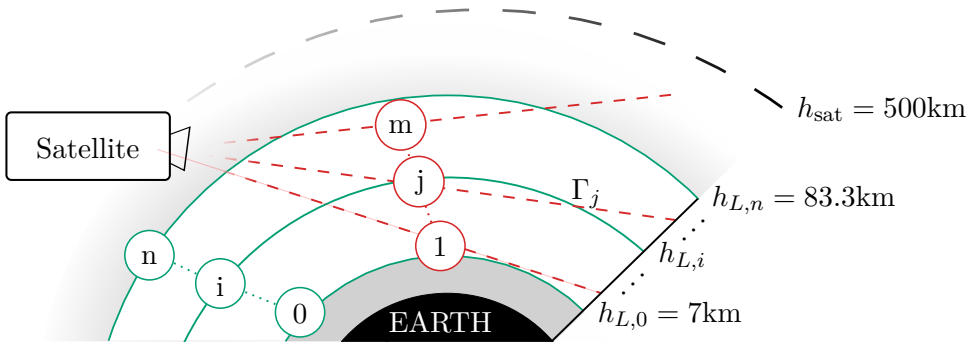

Figure 3.1: Schematic of measurement and analysis geometry, not to scale. The stationary satellite, at a constant height h_{sat} above Earth, takes $m = 41$ measurements along its line-of-sight defining by the line Γ_j. Each measurement has a limb height ℓ_j, $j = 1, 2, \ldots, m$ defined as the closest distance of Γ_j to the Earth surface. Between $h_{L,0} = 7\text{km}$ and $h_{L,n} = 83.3\text{km}$, the stratosphere is discretised into $n = 44$ layers as illustrated by the solid green lines.

Targeting the thermal radiation at one wave number ν of one specific molecule, the j^{th} measurement, is modelled by the radiative transfer equation (RTE) [27]

$$y_j = \int_{\Gamma_j} B(\nu, T) k(\nu, T) \frac{p(T)}{k_{\text{B}} T(r)} x(r) \tau(r) \mathrm{d}r + \eta_j \tag{3.1}$$

$$\tau(r) = \exp\left\{ - \int_{r_{\text{obs}}}^{r} k(\nu, T) \frac{p(T)}{k_B T(r')} x(r') \mathrm{d}r' \right\}, \tag{3.2}$$

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

$$k(\nu, T) = L(\nu, T_{\text{ref}}) \frac{Q(T_{\text{ref}})}{Q(T)} \frac{\exp\{-c_2 E''/T\}}{\exp\{-c_2 E''/T_{\text{ref}}\}} \frac{1 - \exp\{-c_2 \nu / T\}}{1 - \exp\{-c_2 \nu / T_{\text{ref}}\}} \tag{3.3}$$

is depend on the line intensity $L(\nu, T_{\text{ref}})$ at reference temperature $T_{\text{ref}} = 296K$, the lower-state energy of the transition E'', the second radiation constant $c2 = 1.4387769\text{cmK}$ all provided by the HITRAN database [28]. The total internal partition function for the lower-state energy is

$$Q(T) = g'' \exp\{-\frac{c_2 E''}{T}\}, \tag{3.4}$$

with the statistical weight g'' (also called the degeneracy factor) accounting for the molecules non-rotational and rotational energy states, see [29]. Under the assumption of local thermodynamic equilibrium (LTE) the black body radiation act as a source function

$$B(\nu, T) = \frac{2hc^2\nu^3}{\exp\{\frac{hc\nu}{k_B T}\} - 1}, \tag{3.5}$$

with Planck's constant h and velocity of light c [**<empty citation>**]. For fundamentals on the Radiative transfer equation we recommend [30, 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 $h_{L,i-1} < h_{L,i}$, for $i = 1, \ldots, n$, with h_0 and $h_{L,n}$. Then we can discretise the ozone, pressure and temperature profiles as a function of height, where in between the heights $h_{L,i-1}$ and $h_{L,i}$, 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 $h_{L,n} =$ and below $h_{L,0} =$, 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 $\boldsymbol{x} = \{x_1, x_2, \ldots, x_n\} \in \mathbb{R}^n$ or the fraction of pressure and temperature $\boldsymbol{p/T} = \{p_1/T_1, p_2/T_2, \ldots, p_n/T_n\} \in \mathbb{R}^n$, we solve the integral

in Eq. (3.1) using the trapezoidal rule so that we can rewrite the integral to a vector multiplication $\boldsymbol{A_j}(\boldsymbol{x}, \boldsymbol{p}, \boldsymbol{T})\,\boldsymbol{x}$ or $\boldsymbol{A_j}(\boldsymbol{x}, \boldsymbol{p}, \boldsymbol{T})\,\boldsymbol{p}/\boldsymbol{T}$, 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

$$\boldsymbol{y} = \boldsymbol{A}(\boldsymbol{x}, \boldsymbol{p}, \boldsymbol{T})\,\boldsymbol{x} + \boldsymbol{\eta} = \boldsymbol{A}(\boldsymbol{x}, \boldsymbol{p}, \boldsymbol{T})\,\frac{\boldsymbol{p}}{\boldsymbol{T}} + \boldsymbol{\eta}\,. \tag{3.6}$$

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 conditioned on $\boldsymbol{x}$ or $\boldsymbol{p}/\boldsymbol{T}$ and we ignore temperature dependencies of the pressure, absorption constant or the black body radiation.

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. (3.2). Then each entry in the row vector $\boldsymbol{A}_{L,j}$ is either defined by $B(\nu)k(\nu)\frac{\boldsymbol{p}}{k_\mathrm{B}\boldsymbol{T}}\mathrm{d}r$ or $B(\nu)k(\nu)\frac{\boldsymbol{x}}{k_\mathrm{B}}\mathrm{d}r$, as in Eq. (3.1). 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.

$h_{L,0}$ does not influences the values for $p_{L,O}/T_{L,0}$ dont include bend of the line integral

4
Results and Conclusions

In this chapter we use the forward model to generate data and explain how we characterise posterior distributions. We compare to a ground truth and draw conclusion about how inforamtive the data is. In doing so we set up a Bayesian framework and guide the reader through the process of prior modelling. We then present results and compare the tensor train approximation of the posterior with samples from posterior distribution the generated by Markov chain Monte-Carlo (MCMC) methods. All programming and analysis is done in python on a MacBook Pro from 2019 with 2.4 Ghz quadcore intel core i5 processor.

4.1 Simulate Data and ground truth

We take a ground truth ozone profile generated from some data [31] of the microwave limb sounder on the aura satellite in the Antarctic region with a peak in high altitude to show that the data is uninformative in those regions, see Fig. 4.9.

From [31] we get ozone volume mixing ratios and pressure tuples. We connect pressure and height values recursively with the hydrostatic equilibrium equation

$$\frac{\mathrm{d}p}{p} = \frac{-gM}{R^*T}\mathrm{d}h\,, \tag{4.1}$$

with the acceleration due to gravity

$$g = g_0\left(\frac{r_0}{r_0+h}\right), \tag{4.2}$$

where the polar radius pf the earth is $r_0 \approx 6356\,\mathrm{km}$, the gravitation at sea level is $g_0 \approx 9.81\mathrm{m/s}^2$, $R^* = 8.31432 \times 10^{-3}\mathrm{Nm/kmol/K}$ and the mean molecular weight of the air is set to $M = 28.97\mathrm{kg/kmol}$ [32]. This holds up to a geometric height of 86km, where ignore a 0.04% change in M from 80km to 86km in geometric altitude.

Following [32] we can form a temperature function

$$T(h) = \begin{cases} T_0, & h = 0 \\ T_0 + a_0 h, & 0 \leq h < h_1 \\ T_0 + a_0 h_1, & h_1 \leq h < h_2 \\ T_0 + a_0 h_1 + a_1(h - h_2), & h_2 \leq h < h_3 \\ T_0 + a_0 h_1 + a_1(h_3 - h_2) + a_2(h - h_3), & h_3 \leq h < h_4 \\ T_0 + a_0 h_1 + a_1(h_3 - h_2) + a_2(h_4 - h_3), & h_4 \leq h < h_5 \\ T_0 + a_0 h_1 + a_1(h_3 - h_2) + a_2(h_4 - h_3) + a_3(h - h_5), & h_5 \leq h < h_6 \\ T_0 + a_0 h_1 + a_1(h_3 - h_2) + a_2(h_4 - h_3) + a_3(h_6 - h_5) + a_4(h - h_6), & h_6 \leq h \lesssim 86 \end{cases} \tag{4.3}$$

with values

subscript i	geometric height h_i in km	gradient a_i
0	0	-6.5
1	11	0
2	20.1	1
3	32.2	2.8
4	47.4	0
5	51.4	-2.8
6	71.8	-2

as in [32], we plot the ground truth temperature in Fig. 4.4. isothermal layers we are able to change the thickenss but we accept that they are is thermeonal we do not model 0. Then we can compute a data vector $\boldsymbol{y}$, with $m = 42$ measurements according to the radiative transfer equation 3.1, determined by the satellite pointing accuracy of 150arcsec as requested in [33], within an atmosphere $h_{L,0} = 7\text{km}$ and $h_{L,n} = 83.3\text{km}$ with $n = 45$ layers. The height values $h_{L,i}$ for each layer $i = 0, \ldots, n$ are defined by the given ozone profile and its pressure values. We target thermal radiation at a wave number $\nu = 7.86\text{cm}^{-1}$, equal to a frequency of roughly 235GHz, where we assume that ozone is the only emitter and calculate the absorption constant $k(\nu, T)$ according to the *HITRAN* database [28] for the isotopologue $^{16}O_3$ with the AFGL Code 666. Lastly we add normally distributed $\boldsymbol{\nu} \sim \mathcal{N}(0, \gamma^{-1}\boldsymbol{I})$ noise so that we have a voltage Signal-to-Noise (SNR) of 60, similar to THz module on the MLS aura satellite [34].

To the pressure values in between $h_{L,0} = 7\text{km}$ and $h_{L,n} = 83.3\text{km}$ we can fit an exponential function

$$p(h) = \exp\{-b\,(h - h_0)\}\, p_0\,, \tag{4.4}$$

with the gradient b and the tuple (p_0, h_0), which will serve as our ground truth pressure function as plotted in Fig. 4.5.

4.2 Set up the Bayesian framework

Given the data and the forward model we set up our Bayesian framework. In doing so we first draw a directed acyclic graph (DAG) to visualise the measurement and modelling process and determine correlations within some prior distribution are already determined. Then we define prior distribution over all parameters, which then leads us to the posterior distributions including a likelihood.

We draw a DAG for the measurement and modelling process, where put the hyper-parameters $\gamma, \delta, h_0, p_0, b, \boldsymbol{h}, \boldsymbol{T}_0, \boldsymbol{a}$ on the top and parameters $\boldsymbol{x}, \boldsymbol{p}/\boldsymbol{T}$ further down. These parameters go into the forward model $\boldsymbol{A}_{NL}$ and generate some noisy data $\boldsymbol{y}$, where the noise is described through the hyper-parameter γ, from space of all measurable noise free data $\boldsymbol{\Omega}$.

The hyper-parameter related to the ozone is δ, describing the smoothness of the ozone profile $\boldsymbol{x}$, and determines the distribution over the ozone profiles, which we set to be a normal distribution, see section 2.2, as we like to deal with a linear-Gaussian inverse problem.

Since we can describe pressure $\boldsymbol{p}$ and temperature $\boldsymbol{T}$ through the functions in Eq. 4.3 and 4.4 we include their function parameters as hyper-parameters in Fig. 4.1. Then we can already see the correlation between pressure and temperature since it goes like $\boldsymbol{p}/\boldsymbol{T}$ into the forward model.

say that we treat ozone and pressure and temperature separately

$$\pi(h_1, p_0, b_1, b_2, \boldsymbol{h_T}, \boldsymbol{c_T}, \boldsymbol{a_T} | \boldsymbol{y}, \gamma, \boldsymbol{x}) \propto \exp\Big\{ -\frac{\gamma}{2} \left\| \boldsymbol{y} - \boldsymbol{A}\frac{\boldsymbol{p}}{\boldsymbol{T}} \right\|^2 + \ln \pi(h_1, p_0, b_1, b_2, \boldsymbol{h_T}, \boldsymbol{c_T}, \boldsymbol{a_T}) + c \Big\}, \tag{4.5}$$

Ideally we like to determine the joint posterior

$$\pi(h_1, p_0, b_1, b_2, \boldsymbol{h_T}, \boldsymbol{c_T}, \boldsymbol{a_T}, \delta, \gamma, \boldsymbol{x} | \boldsymbol{y}) \tag{4.6}$$

but this is computaional not feasible

with normally distributed likelihood function.

4.2.1 Prior Modelling

To complete the Bayesian framework we have define prior distribution for the hyper-parameters and parameters. Ideally we define the prior distributions as uninformative as possible within functional dependencies and valid physical properties. In this section we describe how we find the priors and what the priors can already tell us about the results. We summarise all prior distributions in Tab. 4.1 and plot the priors for the hyper-parameters in Fig. 4.12 and 4.16 to 4.20 as a dotted black line and for the parameters in Fig. 4.2 to 4.5.

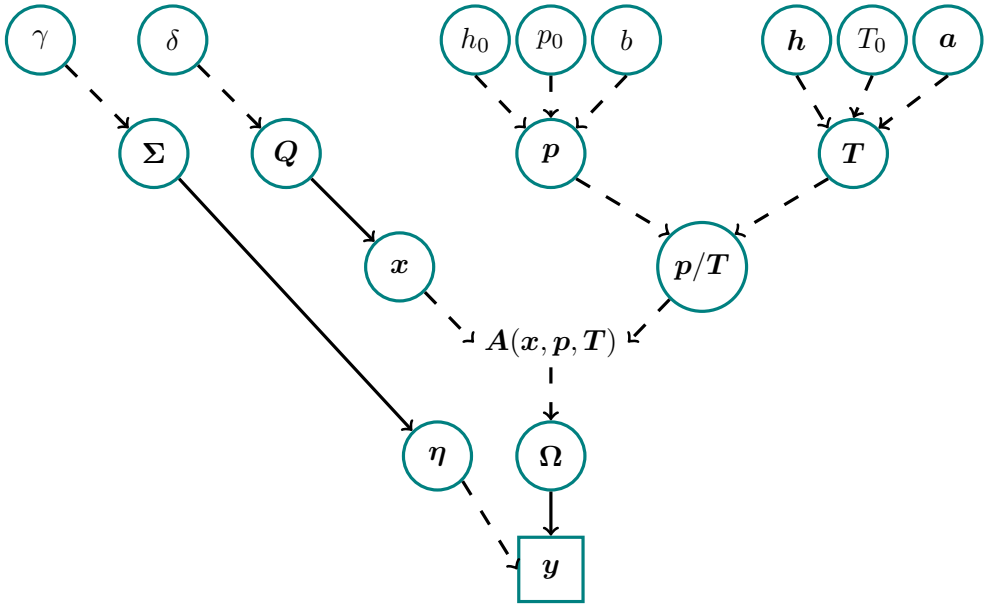

Figure 4.1: Complete directed acyclic graph of the forward model. The hyper-parameters at the top deterministically (dotted line) describe the parameters $(\boldsymbol{p}/\boldsymbol{T})$ or the noise covariance $\boldsymbol{\Sigma} = \gamma^{-1}\boldsymbol{I}$ of the random (solid line) noise $\boldsymbol{\eta} \sim \mathcal{N}(0, \gamma^{-1}\boldsymbol{I})$ and precision matrix $\boldsymbol{Q} = \delta\boldsymbol{L}$ of the distribution of $\boldsymbol{x} \sim \mathcal{N}(0, \delta\boldsymbol{L})$, where $\boldsymbol{L}$ is a graph Laplacian as in Eq. 4.7. We can group the noise precision γ and the smoothness parameter δ to define the marginal posterior over those hyper-parameters and then condition on them for the conditional posterior distribution,for further details see Fig. 4.6. In this whole process where we condition on the pressure $\boldsymbol{p}$ and temperature $\boldsymbol{T}$, which we retrieve separately, see Fig. 4.15. The hyper-parameters h_0, p_0, b deterministically describe the pressure function in Eq. 4.4, note that we only need three parameters here since $h_0 < h_{L,0}$ and $\boldsymbol{h} = \{h_1, h_2, h_3, h_4, h_5, h_6\}$, $\boldsymbol{a} = \{a_0, a_1, a_2, a_3, a_4\}$ and T_0 determine the temperature function. The parameters $\boldsymbol{x}$ and $\boldsymbol{p}/\boldsymbol{T}$ determine the space of all measurable noise free data $\boldsymbol{\Omega}$ through the forward model $\boldsymbol{A}(\boldsymbol{x}, \boldsymbol{p}, \boldsymbol{T})$ from which we randomly observe data set plus some random noise.

Ozone

The priors for $\gamma \sim \mathcal{T}(1, 10^{-10})$ and $\delta \sim \mathcal{T}(1, 10^{-10})$ are gamma distribution, with parameters chosen so that the distributions are relatively uninformative, see black line in Fig. 4.12. another reason to choose a gamma distribution is that then the marginal posterior for $\pi(\gamma|\delta, \boldsymbol{y})$ is gamma distribution as well and easy to sample from, conjugate prior.

The hyper-parameter delta is defining a prior distribution for ozone $\boldsymbol{x} \sim \mathcal{N}(0, \delta\boldsymbol{L})$, with

$$\delta\boldsymbol{L} = \delta \begin{bmatrix} 2 & -1 & & & \\ -1 & 2 & -1 & & \\ & \ddots & \ddots & \ddots & \\ & & -1 & 2 & -1 \\ & & & -1 & 2 \end{bmatrix} \tag{4.7}$$

which is graph Laplacian with Dirichlet boundary conditions [35]. We plot the prior distribution of ozone in Fig. 4.2 and see that it is quite uninformative, due to the

		TT bounds			
model parameters	priors	lower	upper	τ_{int}	Context
γ	$\mathcal{T}(1, 10^{-10})$	$5\,10^{-8}$	$4.5\,10^{-7}$		$\boldsymbol{y}$
δ	$\mathcal{T}(1, 10^{-10})$	-	-		$\boldsymbol{x}$
λ	-	500	7000		$\boldsymbol{x}$
$\boldsymbol{x}$	$\mathcal{N}(0, \delta \boldsymbol{L})$	-	-		$\boldsymbol{x}$
h_0	$\mathcal{N}(5.5, 0.5)$	4.76	5.74		$\boldsymbol{p/T}$
p_0	$\mathcal{N}(500, 6)$	479	519		$\boldsymbol{p/T}$
b	$\mathcal{N}(0.167, 7\,10^{-4})$	0.165	0.170		$\boldsymbol{p/T}$
h_1	$\mathcal{N}(11, 0.1)$	10.6	11.3		$\boldsymbol{p/T}$
h_2	$\mathcal{N}(20.1, 0.9)$	16.7	22.8		$\boldsymbol{p/T}$
h_3	$\mathcal{N}(32.3, 3)$	23.8	43.6		$\boldsymbol{p/T}$
h_4	$\mathcal{N}(47.4, 0.5)$	45.5	49.3		$\boldsymbol{p/T}$
h_5	$\mathcal{N}(51.4, 0.5)$	49.5	53.3		$\boldsymbol{p/T}$
h_6	$\mathcal{N}(71.8, 3)$	60.6	83.1		$\boldsymbol{p/T}$
a_0	$\mathcal{N}(-6.5, 0.01)$	-6.54	-6.46		$\boldsymbol{p/T}$
a_1	$\mathcal{N}(1, 0.01)$	0.96	1.04		$\boldsymbol{p/T}$
a_2	$\mathcal{N}(2.8, 0.1)$	2.43	3.18		$\boldsymbol{p/T}$
a_3	$\mathcal{N}(-2.8, 0.1)$	-3.18	-2.43		$\boldsymbol{p/T}$
a_4	$\mathcal{N}(-2, 0.01)$	-2.04	-1.96		$\boldsymbol{p/T}$
T_0	$\mathcal{N}(288.15, 2)$	281.8	294.5		$\boldsymbol{p/T}$

Table 4.1: Summary of relevant parameter characteristics, bounds and sampling statistics. Gaussian $\mathcal{N}(\mu, \sigma)$ and gamma distribution $\mathcal{T}(\alpha = \text{scale}, \beta = \text{rate})$ Bounds for tt test if would work with previous gamma prior or fix gamma prior with set values

scale we can not really see the true Ozone profile and refer to Fig. 4.9 or Fig. 4.13 for better look at the true ozone profile.

Pressure over temperature

Next we define the normal hyper-priors for the temperature and pressure hyper-parameters as in table 4.1, which lead to temperature and pressure prior samples see Fig. 4.4 and 4.5. When plotting the prior samples as $\boldsymbol{p/T}$ in Fig. 4.3, since that is how they go into the forward model, we can already see that pressure structure is dominating. More specifically in Fig. D.6 that for example the temperature at the lowest atmospheric layer $h_{L,0}$ does not influences the values for $p_{L,O}/T_{L,0}$. We choose the priors for the height values so that they are very unlike to ovelap, see Fig. D.3. Additionally we will chose the TT boundaries so that height values cant overlap, the table 4.1 and results later 4.16 and 4.17.

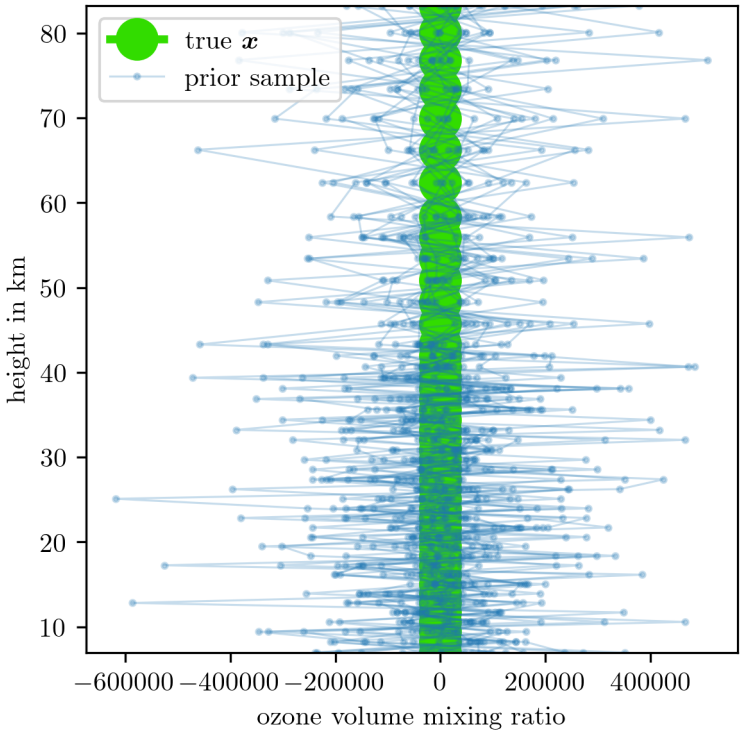

Figure 4.2: We draw samples from ozone prior distribution $\boldsymbol{x} \sim \mathcal{N}(0, \delta \boldsymbol{L})$ after generating a sample from the hyper-prior distribution $\delta \sim \mathcal{T}(1, 10^{-10})$. Note that since the spread/variance of prior samples is very large compared to the ozone volume mixing ratios, the ozone profile appears to be constant, which it is not, as seen e.g. in Fig. 4.9.

Since we can fit an exponential to the pressure above a height of $h_{L,0}$ we do not allow the prior for h_0 to go above that value, see Fig. D.4 and also choose the grid of the TT so that this does not happen. We choose normal prior distribution for b and p, where we restrict ourselves slightly for the TT approximation.

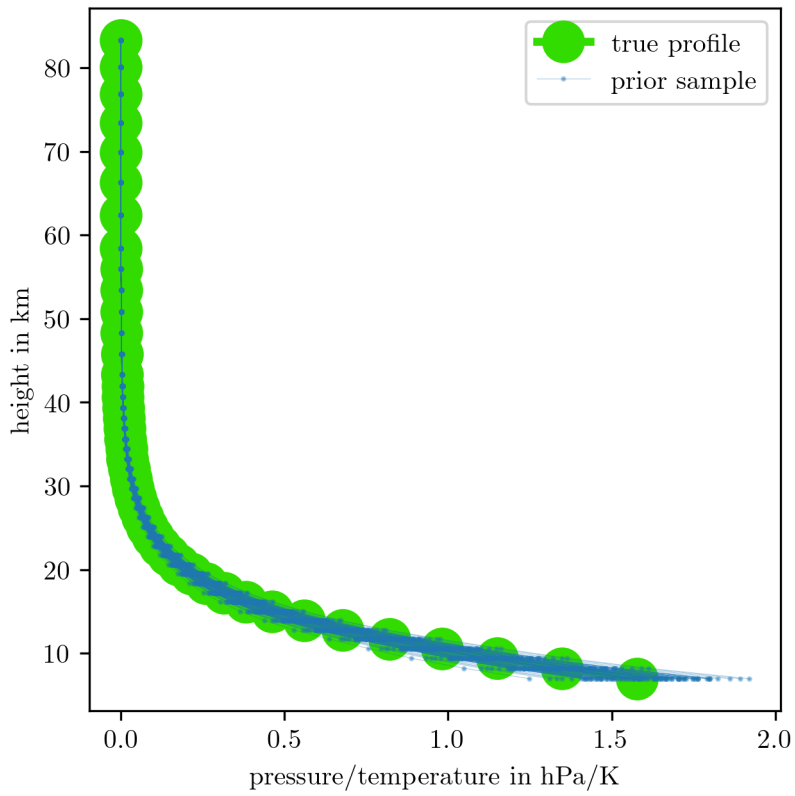

Figure 4.3: We draw samples from the hyper-prior distribution of $h_0, b, p_0, h_1, h_2, h_3, h_4, h_5, h_6, a_0, a_1, a_2, a_3, a_4$ and T_0 as defined in table 4.1 and then calculate $\boldsymbol{p}/\boldsymbol{T}$ according to the functions in Eq. 4.4 and 4.3.

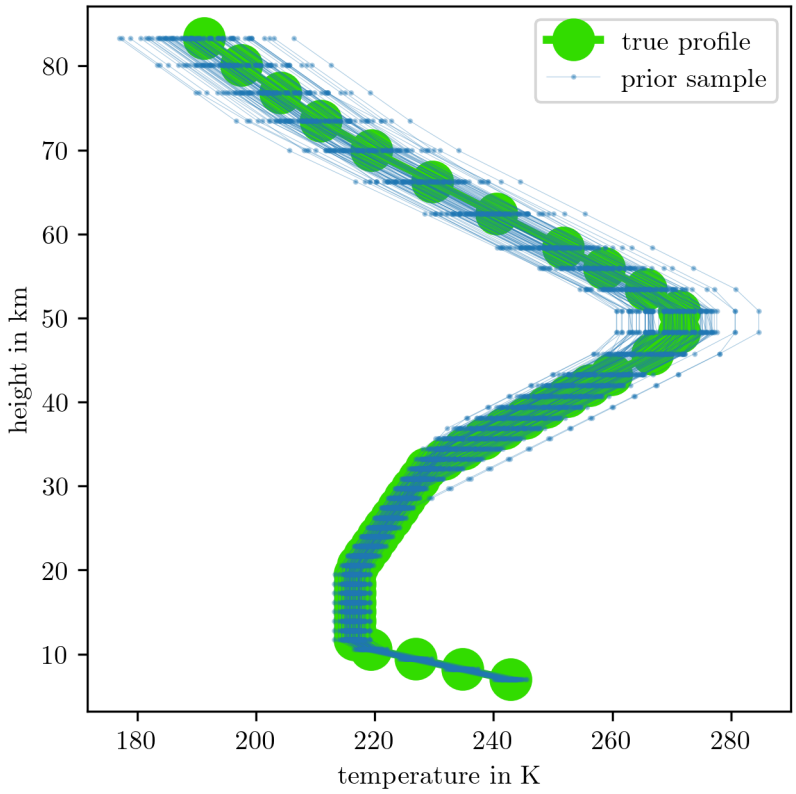

Figure 4.4: We draw samples from the hyper-prior distribution of $h_1, h_2, h_3, h_4, h_5, h_6, a_0, a_1, a_2, a_3, a_4$ and T_0 as defined in table 4.1 and then calculate $\boldsymbol{T}$ according to the function in Eq. 4.3.

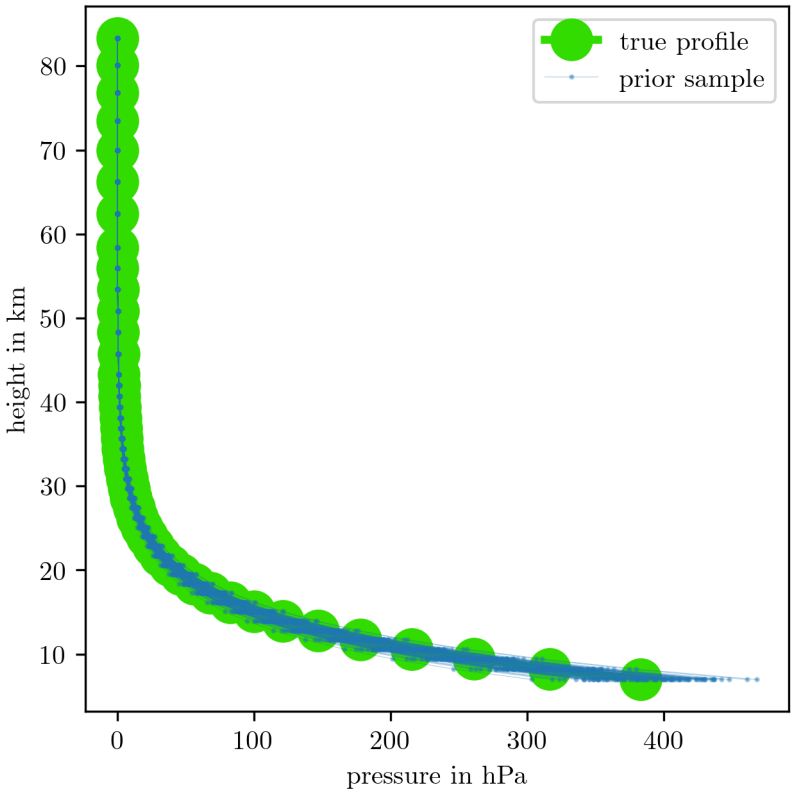

Figure 4.5: We draw samples from the hyper-prior distribution of h_0, b and p_0 as defined in table 4.1 and then calculate $\boldsymbol{p}$ according to the function in Eq. 4.4.

4.3 Posterior distributions with Linear model for Ozone – MTC

In this section we calculate the posterior marginal and then conditional (MTC) posterior distribution for ozone conditioned on the ground truth temperature and pressure profiles using the linear forward model $\boldsymbol{A}_L$. This is faster then the other way round (finding temperature over pressure conditioning on ozone) and temperature and pressure are well defined within the atmosphere so it is easier to just condition on a temperature and pressure profile out of a text book. We employ a so-called Metropolis within Gibbs (MWG) algorithm on the marginal posterior as summarised in the algorithmic Box 2 or use a Tensor-Train (TT) approximation to calculate marginal posterior values. Then we can either sample from the conditional posterior using the randomise then optimise (RTO) method or calculate conditional mean and variance using quadrature.

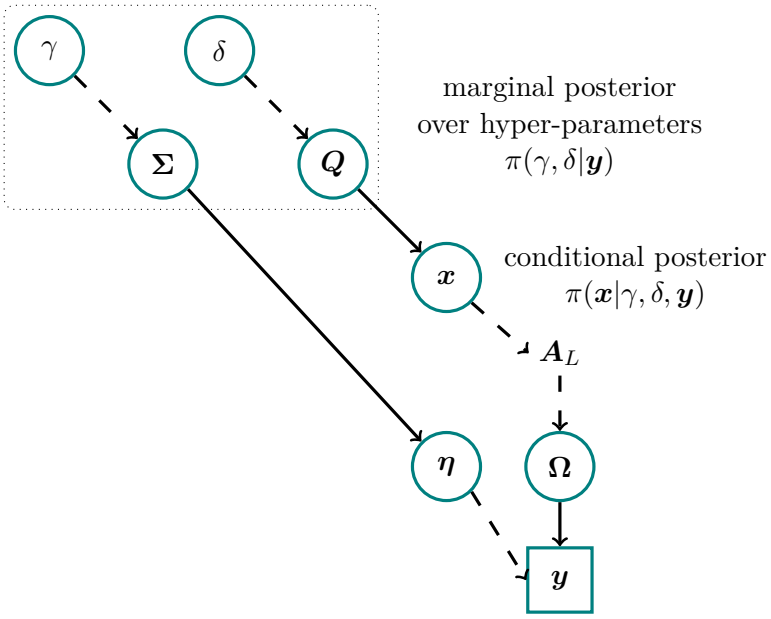

Figure 4.6: Directed acyclic graph for ozone retrieval and MTC scheme as described in Fig. 4.1. The hyper-parameters δ and γ determine the noise covariance $\boldsymbol{\Sigma}$ for the random noise vector $\boldsymbol{\eta} \sim \mathcal{N}(0, \gamma^{-1}\boldsymbol{I})$ and the prior precision matrix $\boldsymbol{Q} = \delta\boldsymbol{L}$ for the distribution over $\boldsymbol{x} \sim \mathcal{N}(0, \delta\boldsymbol{L})$, where $\boldsymbol{L}$ is the graph Laplacian, see Eq. 4.7. In the MTC scheme we evaluate the marginal posterior over the hyper-parameters $\pi(\gamma, \delta|\boldsymbol{y})$ as in Eq. **??** first and then conditional posterior $\pi(\boldsymbol{x}|\gamma, \delta, \boldsymbol{y})$ as in Eq. **??**. The parameter$\boldsymbol{x}$ determine the space of all measurable noise free data $\boldsymbol{\Omega}$ through the forward model $\boldsymbol{A}(\boldsymbol{x}, \boldsymbol{p}, \boldsymbol{T})$ from which we randomly observe a data set plus some random noise. Note that once we found an affine map we update the forward model to $\boldsymbol{M}\boldsymbol{A}_L$.

The DAG in Fig. 4.6 visualises that process and we can show explicitly that we group the hyper-parameters δ, γ together to determine the marginal posterior $\pi(\gamma, \delta|\boldsymbol{y})$. Here γ , the noise parameter, determines the noise precision $\boldsymbol{\Sigma} = \gamma^{-1}\boldsymbol{I}$ and δ, the smoothness parameter, the precision matrix $\boldsymbol{Q} = \delta\boldsymbol{L}$ of the prior distribution for $\boldsymbol{x}$. Then conditioned on the hyper-parameters the conditional posterior $\pi(\boldsymbol{x}|\gamma, \delta, \boldsymbol{y})$ gives the distribution of

posterior ozone profiles. Note that we use the linear model A_L here as we do not have an approximation to the non-linear model yet and all prior distributions are defined in Table 4.1. The full posterior $\pi(\boldsymbol{x}, \gamma, \delta|\boldsymbol{y}) = \pi(\boldsymbol{x}|\gamma, \delta, \boldsymbol{y})\pi(\gamma, \delta|\boldsymbol{y})$ is given by multiplication of the marginal and conditional posterior densities.

4.3.1 Hyper-parameters samples from the marginal posterior distribution

The marginal posterior distribution

$$\pi(\lambda, \gamma|\boldsymbol{y}) \propto \lambda^{n/2}\gamma^{m/2} \exp\left\{-\frac{1}{2}g(\lambda) - \frac{\gamma}{2}f(\lambda)\right\}\pi(\lambda, \gamma), \tag{4.8}$$

with $\lambda = \delta/\gamma$, and

$$f(\lambda) = \boldsymbol{y}^T\boldsymbol{y} - (\boldsymbol{A}_L^T\boldsymbol{y})^T(\boldsymbol{A}_L^T\boldsymbol{A}_L + \lambda\boldsymbol{L})^{-1}(\boldsymbol{A}_L^T\boldsymbol{y})\,, \tag{4.9a}$$

$$\text{and } g(\lambda) = \log\det(\boldsymbol{A}_L^T\boldsymbol{A}_L + \lambda\boldsymbol{L})\,, \tag{4.9b}$$

for the linear model $\boldsymbol{A}_L$, see Sec. 2.2.1 for the derivation. To calculate function values more efficiently we approximate the function $f(\lambda)$ and $g(\lambda)$ with 3rd order Taylor series around the mode λ_0 of $\pi(\lambda, \gamma|\boldsymbol{y})$, since the functions are well behaved over a large range of λ see Fig. 4.7. The derivatives for the Taylor series are

$$f^{(r)}(\lambda_0) = (-1)^{r+1}r!(\boldsymbol{A}_L^T\boldsymbol{y})^T(\boldsymbol{B}_0^{-1}\boldsymbol{L})^r\boldsymbol{B}_0^{-1}\boldsymbol{A}_L^T\boldsymbol{y} \tag{4.10}$$

$$\text{and } \log g(\lambda) = (\log\lambda - \log\lambda_0)\frac{\log g(\lambda_{\max}) - \log g(\lambda_0)}{\log\lambda_{\max} - \log\lambda_0} + \log g(\lambda_0) \tag{4.11}$$

with $\boldsymbol{B}_0 = \boldsymbol{A}_L^T\boldsymbol{A}_L + \lambda_0\boldsymbol{L}$. We find the mode at the minimum of $-\log\{\pi(\lambda, \gamma|\boldsymbol{y})\}$ using `scipy.optimize.fmin` function and limit the number of function evaluation to 25 and use Cholesky back and forward substitution to calculate values of $g(\lambda)$ and $f(\lambda)$. Additionally, we calculate $\boldsymbol{B}_0^{-1}\boldsymbol{L}$ and $\boldsymbol{B}_0^{-1}\boldsymbol{A}_L^T\boldsymbol{y}$ once more at λ_0 and plot the Taylor approximation within the sampling region in Fig. 4.7.

To characterise the marginal posterior function we employ a Metropolis within Gibbs (MWG) algorithm on $\pi(\lambda, \gamma|\boldsymbol{y})$. In doing so, one may implement a Metropolis random walk on the full conditional

$$\pi(\lambda|\gamma, \boldsymbol{y}) \propto \lambda^{n/2+\alpha_\delta-1} \exp\left\{-\frac{1}{2}g(\lambda) - \frac{\gamma}{2}f(\lambda) - \beta_\delta\gamma\lambda\right\} \tag{4.12}$$

and do a Gibbs steps on

$$\gamma|\lambda, \boldsymbol{y} \sim \Gamma\left(\frac{m}{2} + \alpha_\delta + \alpha_\gamma, \frac{1}{2}f(\lambda) + \beta_\gamma + \beta_\delta\lambda\right) \tag{4.13}$$

to generate marginal posterior samples $(\lambda, \gamma)^{(1)}, \ldots, (\lambda, \gamma)^{(N)} \sim \pi(\lambda, \gamma|\boldsymbol{y})$. Note that, when changing variables from $\delta = \lambda\gamma$ to λ the hyper-prior distribution changes to $\pi(\lambda) \propto \lambda^{\alpha_\delta-1}\gamma^{\alpha_\delta} \exp(-\beta_\delta\lambda\gamma)$, due to $\mathrm{d}\delta/\mathrm{d}\lambda = \gamma$.

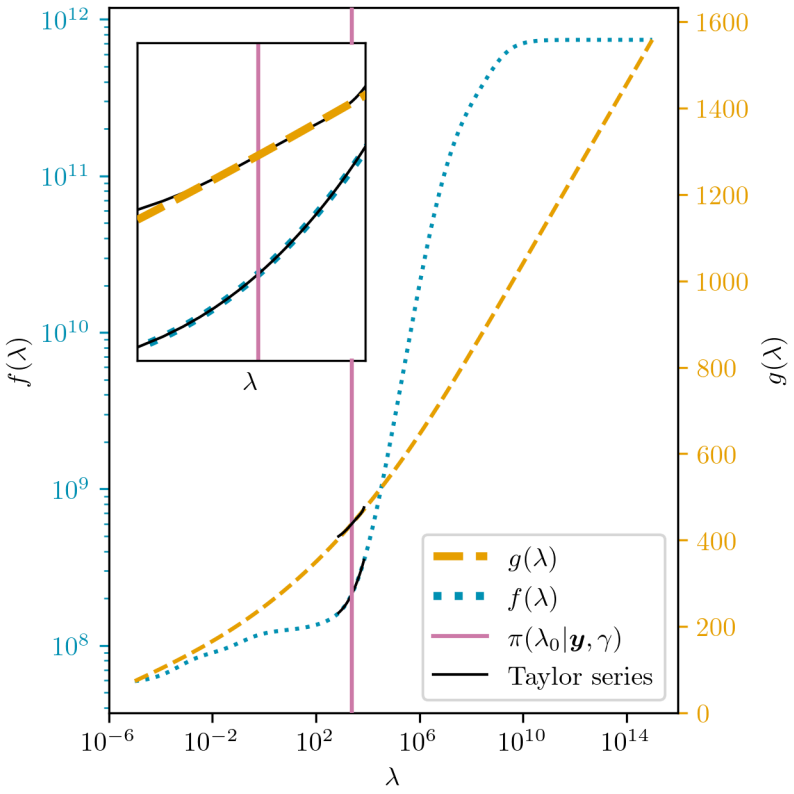

Figure 4.7: Plot of the functions $f(\lambda)$ and $g(\lambda)$ from the marginal posterior for a wide range of $\lambda = \delta/\gamma$. We plot the third order Taylor series in black around the mode of the marginal posterior (vertical line) for the sampling range of λ within the MWG algorithm.

To run a Metropolis random walk on $\pi(\lambda|\gamma,\boldsymbol{y})$ we choose a symmetric proposal distribution $q(\lambda'|\lambda^{(k)}) \sim \mathcal{N}(\lambda^{(k)}, w_\lambda)$ conditioned on the previous sample $\lambda^{(k)}$, with $k = 1,\ldots,N$. Here N is the length of the output chain, but to insure that there are no biases due to the initialisation at the mode $(\lambda^{(0)},\gamma^{(0)}) = (\lambda_0,\gamma_0)$ of the marginal posterior we discard the samples after a burn-in period $N_{\text{burn-in}}$. So the effective output is N - $N_{\text{burn-in}}$. To accept or reject a new sample λ we draw a random uniform number in between 0 and 1 and compare with the acceptance ratio

$$\log\left\{\frac{\pi(\lambda|\gamma^{(t-1)},\boldsymbol{y})}{\pi(\lambda^{(t-1)}|\gamma^{(t-1)},\boldsymbol{y})}\right\} = \log\{\pi(\lambda|\gamma^{(t-1)},\boldsymbol{y})\} - \log\{\pi(\lambda^{(t-1)}|\gamma^{(t-1)},\boldsymbol{y})\} \tag{4.14}$$

$$= \frac{n}{2}(\log\{\lambda\} - \log\{\lambda^{(t-1)}\}) + \frac{1}{2}\Delta g + \frac{\gamma^{(t-1)}}{2}\Delta f + \beta_\delta\gamma^{(t-1)}\Delta\lambda\,, \tag{4.15}$$

which we calculate in the log space. Here $\Delta f = f(\lambda') - f(\lambda^{(k)}) = \sum f^{(r)}(\lambda_0)\Delta\lambda' - \Delta\lambda^{(k)}$, where $\Delta\lambda' = \lambda' - \lambda_0$ and $\Delta\lambda^{(k)} = \lambda^{(k)} - \lambda_0$, and $\Delta g = \exp\log g(\lambda') - \exp\log g(\lambda^{(k)})$.

Lastly, a Gibbs step provides a new $\gamma^{(k+1)} \sim \gamma|\lambda^{(k+1)},\boldsymbol{y}$, see Equation (4.13). See Algorithmic Box 2 for summarised version.

We run the MwG for $N = 20000$ plus $N_{\text{burn-in}} = 100$ steps and set the standard deviation of the normal proposal distribution to $\sigma_\lambda = 0.8\lambda_0$ so that the acceptance rate

Algorithm 2: Metropolis within Gibbs for $\pi(\lambda, \gamma|\boldsymbol{y})$

1: Initialise $\boldsymbol{\theta}^{(0)} = (\lambda^{(0)}, \gamma^{(0)})$ and set burn-in $N_{\text{burn-in}}$
2: **for** $k = 1, \ldots, N'$ **do**
3: Propose $\lambda \sim \mathcal{N}(\lambda^{(t-1)}, 0.8\lambda_0)$
4: Compute

$$\alpha(\lambda|\lambda^{(t-1)}) = \min\left\{1, \frac{\pi(\lambda|\gamma^{(t-1)}, \boldsymbol{y})}{\pi(\lambda^{(t-1)}|\gamma^{(t-1)}, \boldsymbol{y})}\right\}$$

5: Draw $u \sim \mathcal{U}(0, 1)$
6: **if** $\alpha \geq u$ **then**
7: Accept and set $\lambda^{(t)} = \lambda$
8: **else**
9: Reject and keep $\lambda^{(t)} = \lambda^{(t-1)}$
10: **end if**
11: Draw $\gamma^{(t)}|\lambda^{(t)}, \boldsymbol{y} \sim \text{Gamma}\big(0.5\, m + 2, 0.5\, f(\lambda^{(t)}) + 10^{-10}(1 + \lambda^{(t)})\big)$
12: **end for**
13: Output: $(\lambda, \gamma)^{(N_{\text{burn-in}})}, \ldots, (\lambda, \gamma)^{(k)}, \ldots, (\lambda, \gamma)^{(N)} \sim \pi(\lambda, \gamma|\boldsymbol{y})$

is ≈ 0.5 as suggested in [**<empty citation>**]. The samples are plotted in Fig. 4.8 as a 2D scatter plot, as well as the trace of the MwG to show ergodicity.

Alternatively we can approximate the square root of marginal posterior with a Tensor-Train (TT) on a grid as defined in table 4.1 with 40 grid points in each dimension. Here we use the `rect_cross.cross` as a black box algorithm from the `ttpy` python package, based on the rect cross algorithm in [**<empty citation>**]. We set the number of ranks to a constant value equal to four and optimse over those rankes with one sweep. For numerical reasons,to avoid underflow, we have to add a constant $c = 460$, such as $\pi(\lambda|\gamma, \boldsymbol{y}) = \exp\{\log \pi(\lambda|\gamma, \boldsymbol{y}) + c\}$. We calculate the marginals $\pi(\lambda|\boldsymbol{y})$ and $\pi(\gamma|\boldsymbol{y})$ as in section **??**, with a constant $\gamma = 1e - 5$, and plot the TT approximation as a colour code on top of the obtained samples in the scatter plot in Fig. 4.8.

The TT alforith with has nkber of funciton evaluaation set with constant rank r $((D-2)r \times n \times r + 2 \times n \times r)2 \times n_{sweep}$ 400 for TT marg we set assume an absolute approximation error of 1 so that we set $\gamma' = 1/\boldsymbol{\lambda}(\mathcal{X})$

On a MacBook Pro from 2019 with 2.4 Ghz quadcore intel core i5 processor it takes $\lessapprox 0.1$s to find the Tensors to approximate the marginal posterior. In comparisons the to run the MwG takes $\approx 0.7s$ for $N = 20000$ effective samples. integrated autocorrelation time, roughly efficient independent samples

4.3.2 Ozone samples from the conditional posterior

As part of the MTC scheme we draw ozone samples after determining the marginal posterior distribution. In this section we present two ways to draw ozone samples

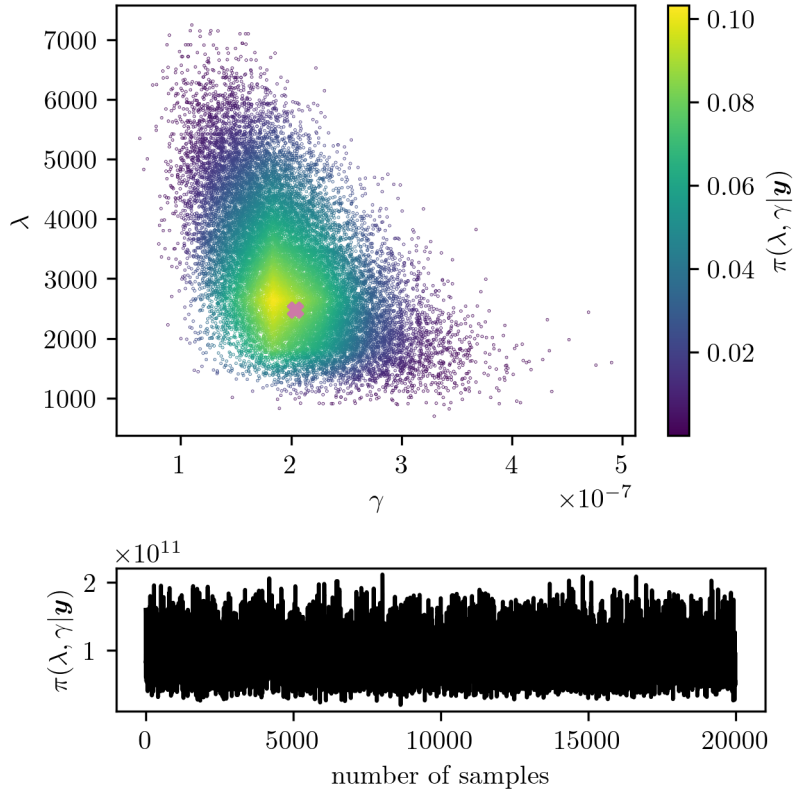

Figure 4.8: We scatter plot the samples of $\lambda = \delta/\gamma$ and γ from the marginal posterior $\pi(\lambda, \gamma|\boldsymbol{y})$ and colour code the samples using the TT approximation of $\pi(\lambda, \gamma|\boldsymbol{y})$. The mode of (λ_0, γ_0) of $\pi(\lambda, \gamma|\boldsymbol{y})$ provided by `scipy.optimize.fmin` is marked with the cross. To show ergodicity we plot the trace of the samples of the Metropolis-within-Gibbs sampler below.

from the conditional posterior

$$\boldsymbol{x}|\delta, \gamma, \boldsymbol{y} \sim \mathcal{N}\big(\underbrace{(\boldsymbol{A}_L^T\boldsymbol{A}_L + \delta/\gamma\boldsymbol{L})^{-1}\boldsymbol{A}_L^T\boldsymbol{y}}_{\boldsymbol{x}_\lambda}, \underbrace{(\gamma\boldsymbol{A}_L^T\boldsymbol{A}_L + \delta\boldsymbol{L})}_{\gamma\boldsymbol{B}_\lambda}{}^{-1}\big)\,. \tag{4.16}$$

First we present the Randomize-than-Optimize (RTO) method [4, 15, 36], as previously described in Sec. 2.4.2. Alternatively we can integrate over the marginal posterior and calculate the mean and variance of the multivariate normal conditional posterior distribution. Note that we reject samples from the conditional posterior with negative ozone values since that is unphysical.

Randomize then optimize – RTO

For the RTO method we start by drawing an independent hyper-parameter sample $(\delta, \gamma) \sim \pi(\delta, \gamma|\boldsymbol{y})$ from the samples of the MwG. Then we generate two independent Gaussian random variables $\boldsymbol{v}_1 \sim \mathcal{N}(\boldsymbol{0}, \gamma\boldsymbol{A}_L^T\boldsymbol{A}_L)$ and $\boldsymbol{v}_2 \sim \mathcal{N}(\boldsymbol{0}, \delta\boldsymbol{L})$. Here can use Cholesky factorisation of $\boldsymbol{L} = \boldsymbol{L}_C\boldsymbol{L}_C^T$ and the multiplication rule for normal distributions

so that $\boldsymbol{v}_1 \sim \sqrt{\gamma}\boldsymbol{A}_L^T\mathcal{N}(0,\boldsymbol{I})$ and $\boldsymbol{v}_2 \sim \sqrt{\delta}\boldsymbol{L}_C\mathcal{N}(0,\boldsymbol{I})$. Then we solve

$$\left(\gamma\boldsymbol{A}_L^T\boldsymbol{A}_L + \delta\boldsymbol{L}\right)\boldsymbol{x} = \gamma\boldsymbol{A}_L^T\boldsymbol{y} + \boldsymbol{v}_1 + \boldsymbol{v}_2\,, \tag{4.17}$$

using Cholesky back and forward substitution, for $\boldsymbol{x}$ and obtain one independent sample of $\pi(\boldsymbol{x}|\boldsymbol{y},\boldsymbol{\theta})$. See Fig. 4.9, where we plot $m =$ samples of the conditional posterior.

The histogram in is binned as we intergate over it to 7 bins

Weighted Mean

Alternatively, we can calculated the mean

$$\mu_{\boldsymbol{x}|\boldsymbol{y}} = \int \boldsymbol{x}_\lambda \pi(\lambda|\boldsymbol{y})\mathrm{d}\lambda \approx \sum \boldsymbol{x}_{\lambda_i}\pi(\lambda_i|\boldsymbol{y})\,, \tag{4.18}$$

and covariance

$$\Sigma_{\boldsymbol{x}|\boldsymbol{y}} = \int \gamma^{-1}\pi(\gamma|\boldsymbol{y})\,\mathrm{d}\gamma \int \boldsymbol{B}_\lambda^{-1}\,\pi(\lambda|\boldsymbol{y})\,\mathrm{d}\lambda \approx \sum {\gamma_i}^{-1}\pi(\gamma_i|\boldsymbol{y}) \sum \boldsymbol{B}_{\lambda_i}^{-1}\pi(\lambda_i|\boldsymbol{y}) \tag{4.19}$$

of the conditional posterior $\pi(\boldsymbol{x}|\delta,\gamma,\boldsymbol{y})$, see Eq. 4.16, by quadrature [37, Sec. 2.1] We get function values for the marginal posterior either by binning the samples in Fig. 4.8 into a normalised histogram and use the height of the bars as quadrature weights, e.g. $\pi(\lambda_i|\boldsymbol{y})$, where λ_i is at the centre of each bin, or from the TT approximation of $\sqrt{\pi(\gamma,\delta|\boldsymbol{y})}$, see also Fig. 4.12. The integral can be interpreted as the weighted average, with $\sum \pi(\lambda_i|\boldsymbol{y}) = 1$ for normalised marginal functions.

If using the samples from the MWG algorithm, we start by binning the samples into 3 bins and stop increasing the number of bins by one if the relative error between the previous and the current conditional posterior mean is less than 0.1%. This happens at a bin number of 5 and gives gives a total number of $3+4+5 = 12$ solves of x_λ to find the conditional mean $\mu_{\boldsymbol{x}|\boldsymbol{y}}$. To calculate calculate the covariance matrix $(\gamma\boldsymbol{B}_\lambda)^{-1}$, we use Cholesky forward and backward substitution to invert $\boldsymbol{B}_\lambda$, solve the integral in Eq. 4.19 5 times.

Using the TT approximation we consider every second grid point of the total 40 grid points of $\pi(\lambda|\boldsymbol{y})$ and $\pi(\gamma|\boldsymbol{y})$, which means we solve the integrals in Eq. 4.18 and 4.19 20 times. We plot the mean and variance in Figure 4.9.

4.4 Approximate non-linear forward model with affine Map

Using an affine map to map in between the linear and non-linear forward map we can approximate the non-linear map and treat the inverse problem as a linear inverse problem. In doing so we use the just calculated posterior distribution to generate ozone sample to generate noise free linear and non-linear data examples which we use to find the affine map.

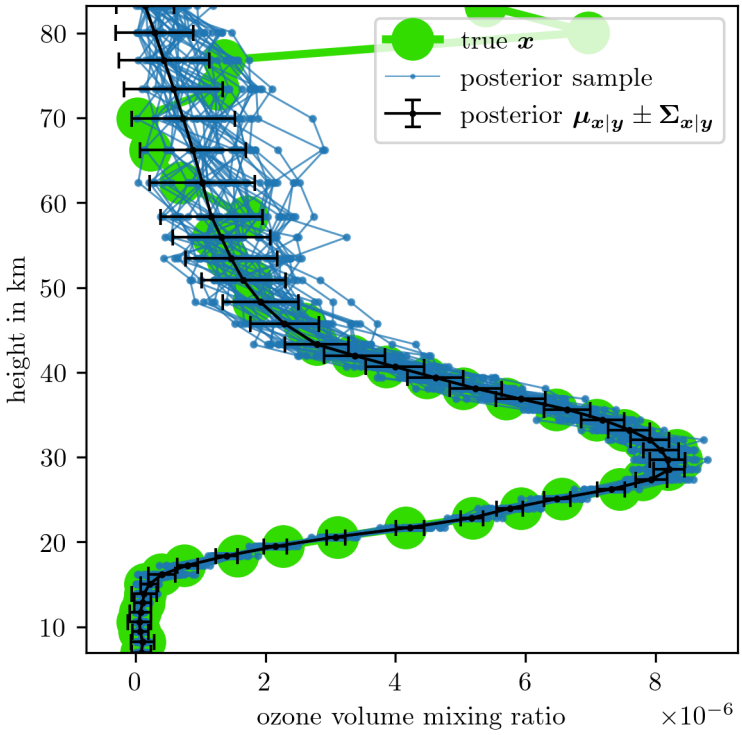

Figure 4.9: We draw samples from the conditional posterior distribution $\pi(\boldsymbol{x}|\lambda, \gamma, \boldsymbol{y})$ after characterising the marginal posterior $\pi(\lambda, \gamma|\boldsymbol{y})$ through sampling or TT approximation using the linear forward map $\boldsymbol{A}_L$. Note that we reject samples with unphysical negative values and effectively treat the conditional posterior as a truncated multivariate normal distribution. We will use those samples to find the affine map $\boldsymbol{M}$, see section 4.4

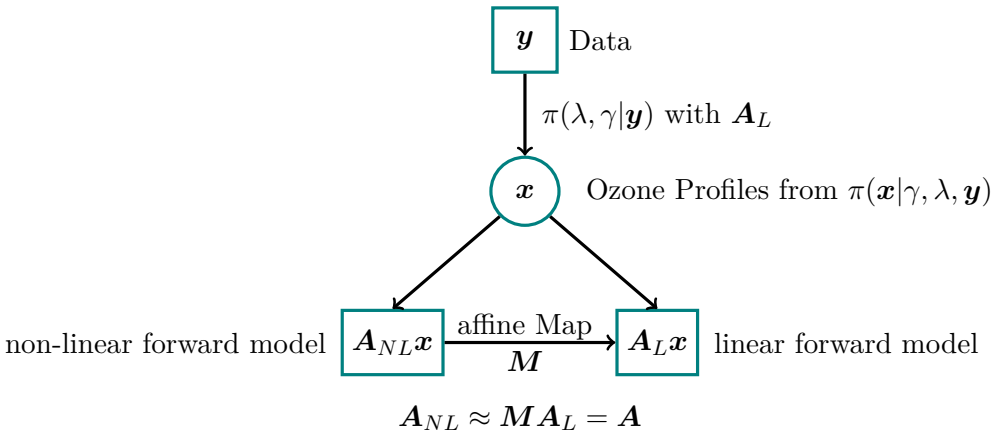

Figure 4.10: The strategy to find the affine map consist of evaluating the marginal posterior for ozone using the linear forward model. Then we draw ozone samples from the conditional posterior and calculate noise free data based on the linear and non-linear forward model. Next we find a mapping in between those two space so that we can approximate the non-linear forward model using an affine map and the linear forward model.

With the samples $\{\boldsymbol{x}^{(1)}, \ldots, \boldsymbol{x}^{(m)}\} \sim \pi(\boldsymbol{x}|, \gamma, \lambda, \boldsymbol{y})$ from the conditional posterior we generate two affine subspaces

$$\boldsymbol{V} = \begin{bmatrix} | & & | & & | \\ \boldsymbol{A}_{NL}(\boldsymbol{x}^{(1)}) & \cdots & \boldsymbol{A}_{NL}(\boldsymbol{x}^{(j)}) & \cdots & \boldsymbol{A}_{NL}(\boldsymbol{x}^{(m)}) \\ | & & | & & | \end{bmatrix} = \begin{bmatrix} — & v_1 & — \\ & \vdots & \\ — & v_j & — \\ & \vdots & \\ — & v_m & — \end{bmatrix}, \tag{4.20}$$

using the non-linear forward model, and

$$\boldsymbol{W} = \begin{bmatrix} | & & | & & | \\ \boldsymbol{A}_L\boldsymbol{x}^{(1)} & \cdots & \boldsymbol{A}_L\boldsymbol{x}^{(j)} & \cdots & \boldsymbol{A}_L\boldsymbol{x}^{(m)} \\ | & & | & & | \end{bmatrix} \tag{4.21}$$

based on the linear forward model, which are matrices in $\mathbb{R}^{m\times m}$. To find the affine map

$$\boldsymbol{M} = \begin{bmatrix} — & r_0 & — \\ & \vdots & \\ — & r_j & — \\ & \vdots & \\ — & r_m & — \end{bmatrix} \in \mathbb{R}^{m\times m}, \tag{4.22}$$

we use the `numpy.linalg.solve` python function to solve

$$v_j = r_j\boldsymbol{W} \tag{4.23}$$

where r_j is the j-th row of $\boldsymbol{M}$.

Finally, we approximate the non-linear forward model

$$\boldsymbol{A}_{NL} \approx \boldsymbol{M}\boldsymbol{A}_L = \boldsymbol{A}\,, \tag{4.24}$$

with the affine map $\boldsymbol{M}$ and the linear forward model $\boldsymbol{A}_L$. In Fig. 4.11 we asses the affine map using one of the samples $\boldsymbol{x} \sim \pi(\boldsymbol{x}|, \gamma, \lambda, \boldsymbol{y})$ from the conditional posterior and calculate the relative error $||\boldsymbol{M}\boldsymbol{A}_L\boldsymbol{x} - \boldsymbol{A}_{NL}\boldsymbol{x}||/||\boldsymbol{M}\boldsymbol{A}_L\boldsymbol{x}||$ in percent between the mapped noise free data and the noise free data based on the non-linear forward. As displayed in Fig. 4.11 we can approximate the non-linear forward model well within the relative difference between the noisy data and noise free non-linear data which is approximately 1.7% and from here on will use the approximated forward map.

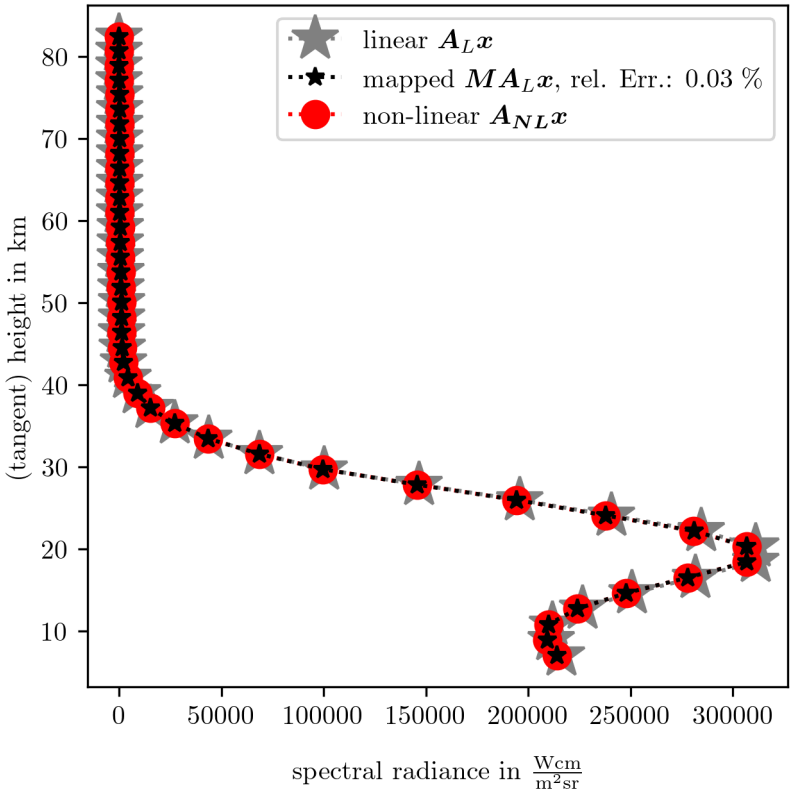

Figure 4.11: We asses how good we can map the linear forward model onto the non-linear forward model using the previous calculated affine map. The gray stars represent noise free linear data, where as the red circles present noise free non-linear data. Then we map the linear noise free data onto the non-linear noise free data and give the relative error in between the mapped noise free data and the non-linear data.

4.5 Posterior distributions with approximated non-linear model for Ozone – MTC

From here on we use the approximation

$$\boldsymbol{A} = \boldsymbol{M}\boldsymbol{A}_L \tag{4.25}$$

of the non-linear forward map to be able to treat the problem as a linear inverse problem. We use the exact same setup as in Sec. 4.3 but with the updated forward map $\boldsymbol{A}$.

4.5.1 Hyper-parameters samples from the marginal posterior distribution

With the updated forward map the marginal posterior distribution becomes

$$\pi(\lambda, \gamma|\boldsymbol{y}) \propto \lambda^{n/2}\gamma^{m/2} \exp\left\{-\frac{1}{2}g(\lambda) - \frac{\gamma}{2}f(\lambda)\right\}\pi(\lambda, \gamma), \tag{4.26}$$

with $\lambda = \delta/\gamma$, and

$$f(\lambda) = \boldsymbol{y}^T\boldsymbol{y} - (\boldsymbol{A}^T\boldsymbol{y})^T(\boldsymbol{A}^T\boldsymbol{A} + \lambda\boldsymbol{L})^{-1}(\boldsymbol{A}^T\boldsymbol{y})\,, \tag{4.27a}$$

$$\text{and } g(\lambda) = \log\det(\boldsymbol{A}^T\boldsymbol{A} + \lambda\boldsymbol{L})\,. \tag{4.27b}$$

Again, we approximate the function $f(\lambda)$ and $g(\lambda)$ with 3rd order Taylor series around the mode λ_0 of $\pi(\lambda, \gamma|\boldsymbol{y})$, provided by `scipy.optimize.fmin`. The Taylor derivatives are

$$f^{(r)}(\lambda_0) = (-1)^{r+1}r!(\boldsymbol{A}^T\boldsymbol{y})^T(\boldsymbol{B}_0^{-1}\boldsymbol{L})^r\boldsymbol{B}_0^{-1}\boldsymbol{A}^T\boldsymbol{y} \tag{4.28}$$

$$\text{and } g^{(r)}(\lambda_0) = (-1)^{r+1}\operatorname{tr}((\boldsymbol{B}_0^{-1}\boldsymbol{L})^r) \tag{4.29}$$

with $\boldsymbol{B}_0 = \boldsymbol{A}^T\boldsymbol{A} + \lambda_0\boldsymbol{L}$.

We run the MWG algorithm and plot the samples in Fig. 4.12 as well as the marginal approximations provided by the TT decomposition.

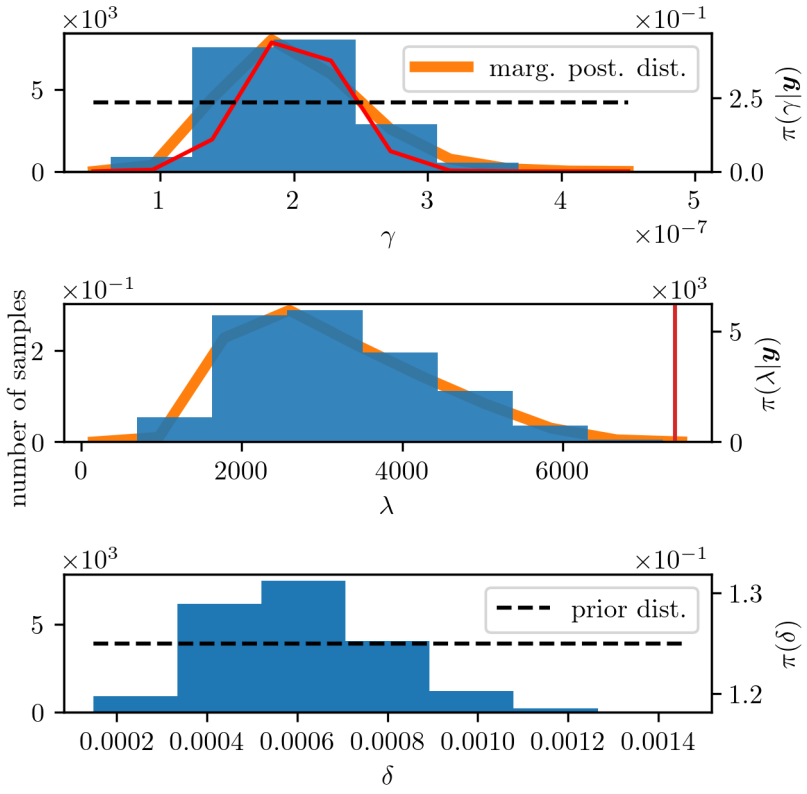

Figure 4.12: We plot the TT approximation of marginal posterior in orange and the samples as a histogram as well as the prior distribution with a dotted line. Note that we sample λ and γ using the Metropolis-within-Gibbs sampler and can calculate δ for every sample of the marginal posterior, we can not do this for the TT approximation. The regularised parameter corresponding to the regularised solution is marked thought the red vertical line at $\lambda_{\text{reg}} =$.

4.5.2 Ozone samples from the conditional posterior and regularised solution

Next, we generate samples from the updated conditional posterior

$$\boldsymbol{x}|\delta,\gamma,\boldsymbol{y} \sim \mathcal{N}(\underbrace{(\boldsymbol{A}^T\boldsymbol{A}+\delta/\gamma\boldsymbol{L})^{-1}\boldsymbol{A}^T\boldsymbol{y}}_{\boldsymbol{x}_\lambda}, \underbrace{(\gamma\boldsymbol{A}^T\boldsymbol{A}+\delta\boldsymbol{L})}_{\gamma\boldsymbol{B}_\lambda}{}^{-1}) \tag{4.30}$$

as described in Section 4.3.2.

We plot the conditional mean and variance in Fig. 4.13 and the regularised solution as well as one sample from the posterior.

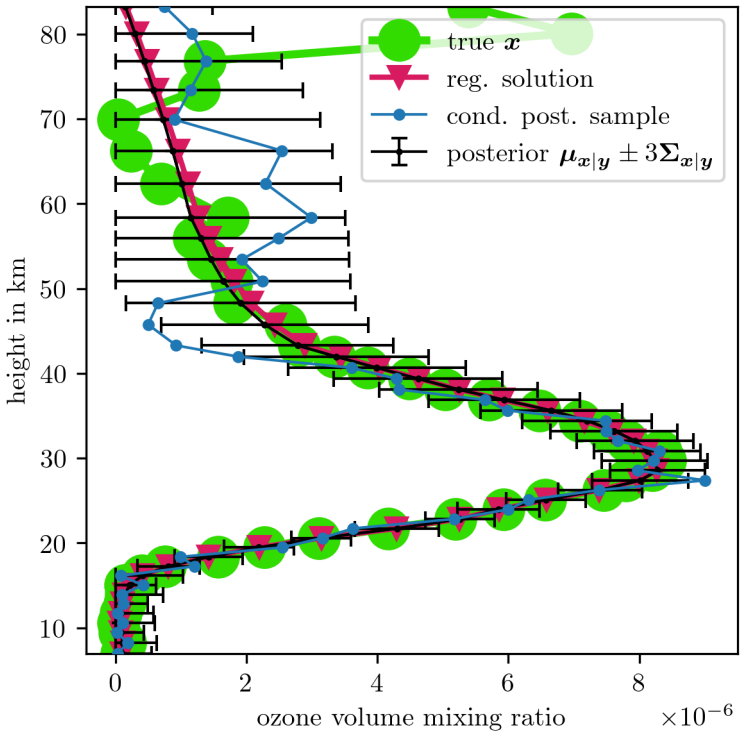

Figure 4.13: We plot the conditional posterior mean and variance in black and the regularised solution on top of the ground truth ozone profile in green. We use the updated forward map $\boldsymbol{M}\boldsymbol{A}_L$

Data uninformative in higher altitude and say again reg sol. vs mean and one samples is good solution

4.5.3 Solution by regularisation

Additionally, we calculate a solution by Tikhonov regularisation as this is most similar to our chosen linear-Gaussian Bayesian framework. The Tikhonov regularised

solution is defined as [38]

$$\boldsymbol{x}_\lambda = \arg\min_{\boldsymbol{x}} \|\boldsymbol{A}\boldsymbol{x} - \boldsymbol{y}\|_2^2 + \lambda \boldsymbol{x}^T \boldsymbol{L}\boldsymbol{x}\,, \tag{4.31}$$

with the regularisation parameter $\lambda = \delta/\gamma$. This maximises the full conditional distribution for $\boldsymbol{x}$, so is not, as often erroneously stated, the maximum a posteriori (MAP) estimate which includes at the hyper-parameters. The regularised solution is typically calculated by solving the normal equations, see Eq. 2.24,

$$\boldsymbol{x}_\lambda = (\boldsymbol{A}^T\boldsymbol{A} + \lambda \boldsymbol{L})^{-1}\boldsymbol{A}^T\boldsymbol{y}\,. \tag{4.32}$$

In doing so we find the best regularisation parameter using the L-curve method [13]. Within this method we compute $\boldsymbol{x}_\lambda$, see Equation (4.32), for 200 different λ values in between 1 to 10^7 and plot the solution semi norm $\sqrt{\boldsymbol{x}_\lambda^T \mathbf{L} \boldsymbol{x}_\lambda}$ against the data misfit norm $\|\boldsymbol{A}\boldsymbol{x}_\lambda - \boldsymbol{x}\|$, see Figure 4.14. The best regularised solution corresponding to the corner of the L-curve is located at the point of maximum curvature, see triangle in Fig. 4.14, which we find with the kneedle algorithm [39] using the python function `kneed.KneeLocator`. This takes roughly seconds on a MacBook Pro from 2019 with 2.4 Ghz quadcore intel core i5 processor.

Sol Reg vs Mean, vs samples, see L-Curve

4.6 Posterior pressure and temperature

To find the posterior pressure and temperature we condition on a γ sample, by fitting a normal distribution to either the samples or the TT approximation of $\pi(\gamma|\boldsymbol{y})$ plotted as a red line in Fig. 4.12, and the ozone sample plotted in 4.13, which is also used to find the affine map. Consequently we use the updated forward map $\boldsymbol{A} = \boldsymbol{M}\boldsymbol{A}_L$ and the posterior is given by

$$\pi(h_1, p_0, b_1, b_2, \boldsymbol{h_T}, \boldsymbol{c_T}, \boldsymbol{a_T}|\boldsymbol{y}, \gamma, \boldsymbol{x}) \propto \exp\Big\{ -\frac{\gamma}{2}\left\|\boldsymbol{y} - \boldsymbol{A}\frac{\boldsymbol{p}}{\boldsymbol{T}}\right\|^2 + \ln \pi(h_1, p_0, b_1, b_2, \boldsymbol{h_T}, \boldsymbol{c_T}, \boldsymbol{a_T}) + c\Big\}\,, \tag{4.33}$$

where c is a constant needed for the TT approximation to avoid underflow.

We define a grid for each of the hyper-parameters, see Tab. 4.1, and set the grid points to 40 and optimise over a set number of ranks equal to the dimension of 15. To define the constant c we evaluate $\frac{\gamma}{2}\left\|\boldsymbol{y} - \boldsymbol{A}\frac{\boldsymbol{p}}{\boldsymbol{T}}\right\|^2 + \ln \pi(h_1, p_0, b_1, b_2, \boldsymbol{h_T}, \boldsymbol{c_T}, \boldsymbol{a_T})$ at 100 randomly chosen grid points and set c to the negative half of the maximum value of those. We terminate the `rect_cross.cross` algorithm from the `ttpy` python package if the relative error provided by the algorithm is smaller than 0.1 or after 10 sweeps, which takes roughly $1-2$ mins on a MacBook Pro from 2019 with 2.4 Ghz quadcore

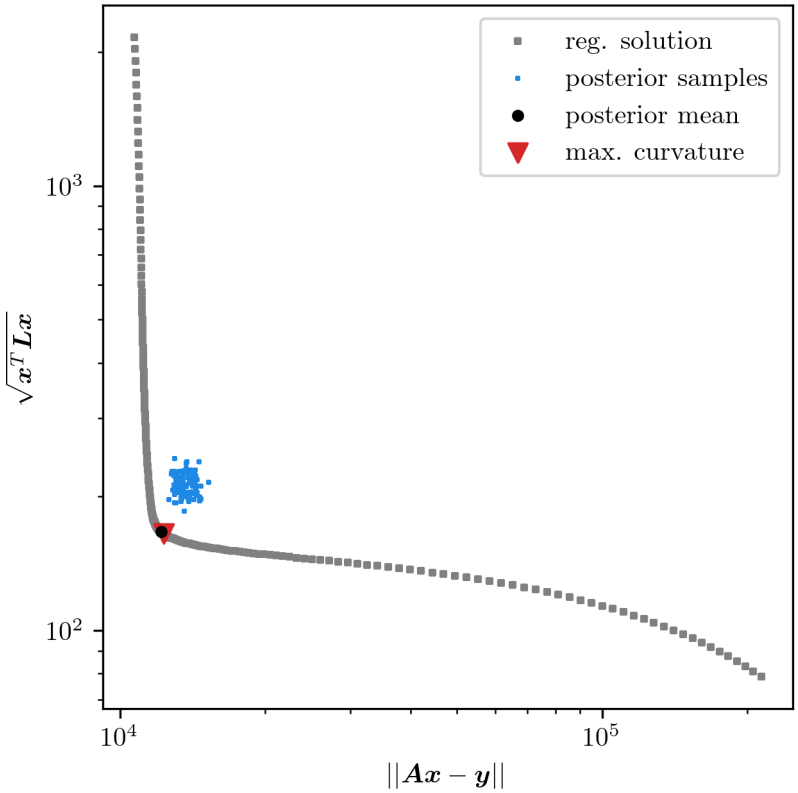

Figure 4.14: We calculate regularised solution as in Eq. **??** and plot the regularised semi norm $\sqrt{\boldsymbol{x}^T\boldsymbol{L}\boldsymbol{x}}$ against the data misfit norm $||\boldsymbol{A}\boldsymbol{x} - \boldsymbol{y}||$ to find the regularised solution at the point of maximum curvature of the so-called L-Curve. Additionally we calculate the data misfit norm and the regularised norm for the ozone posterior and for samples of the conditional posterior distribution. make box around Kneedle reagion

intel core i5 processor. Then we calculate the marginals as in section 2.5 with a constant $= 1e - 15$ and plot the results in Fig. 4.16 to 4.20 as an orange line.

On the same grid we run the t-walk, with the constant set to zero as the t-walk evaluates the function $-\ln \pi(h_1, p_0, b_1, b_2, \boldsymbol{h_T}, \boldsymbol{c_T}, \boldsymbol{a_T}|\boldsymbol{y}, \gamma, \boldsymbol{x})$ we do not run into numerical issues. We download the t-walk from [40] and let it take 1000000 steps plus a burn-in period of 1000, which takes around 7 mins on the same laptop. The resulting histogram are plotted in Fig. 4.16 to 4.20, additionally we plot the trace of the samples in Fig. D.7 integrated autocorrelation time, roughly efficient independent samples

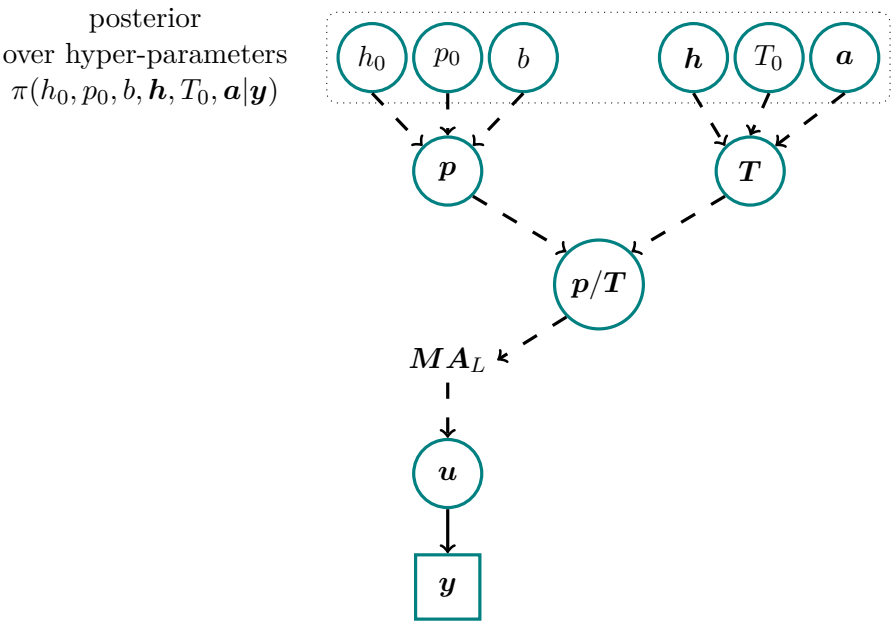

Figure 4.15: Conditioned on an ozone profile the posterior of the hyper-parameters describing pressure and temperature is given as in Eq. **??**. Since pressure and temperature go into the forward model as $\boldsymbol{p/T}$ they are highly correlated but the pressure is the dominant parameter, see Fig. 4.3 and D.6. Note that here we use the updated forward model $\boldsymbol{MA}_L$ and conditioned on a γ sample from the previously evaluated marginal posterior see Fig. 4.12.

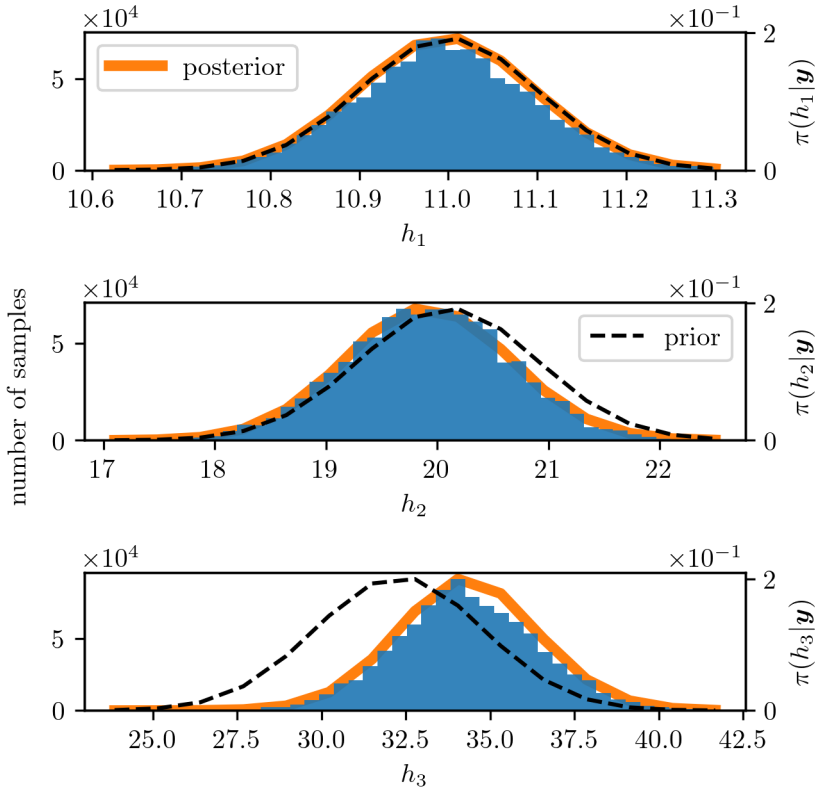

Figure 4.16: We plot the TT approximation of marginal posterior in orange and the samples as a histogram as well as the prior distribution with a dotted line.

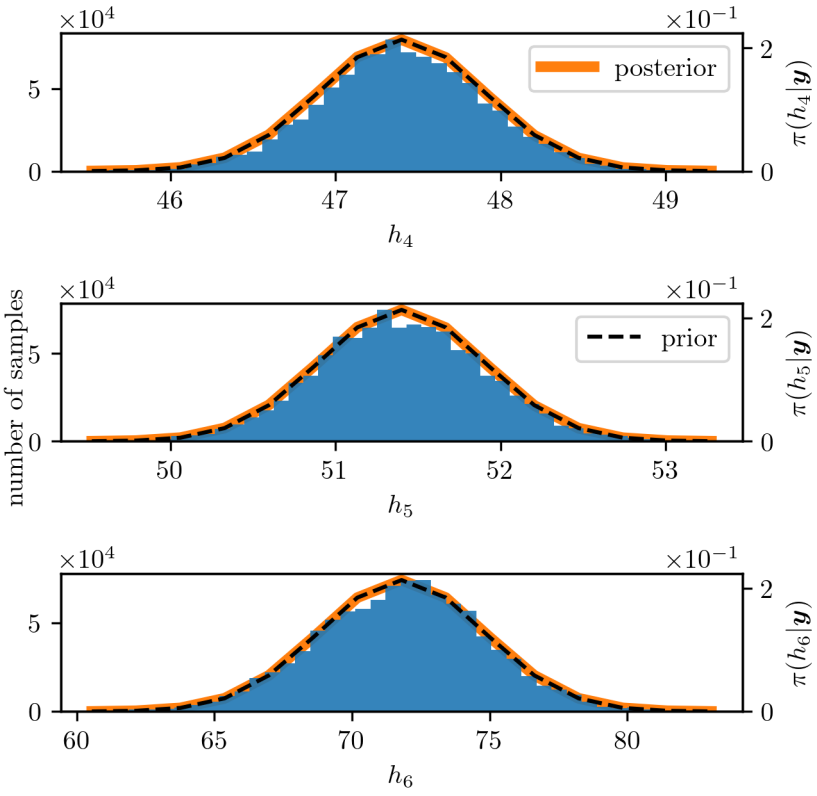

Figure 4.17: We plot the TT approximation of marginal posterior in orange and the samples as a histogram as well as the prior distribution with a dotted line.

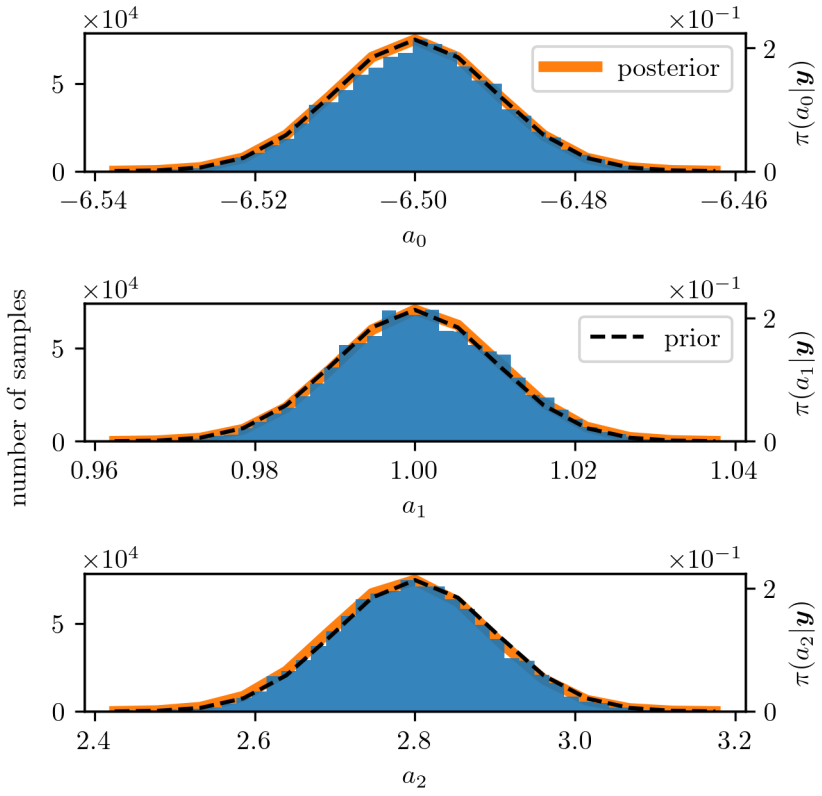

Figure 4.18: We plot the TT approximation of marginal posterior in orange and the samples as a histogram as well as the prior distribution with a dotted line.

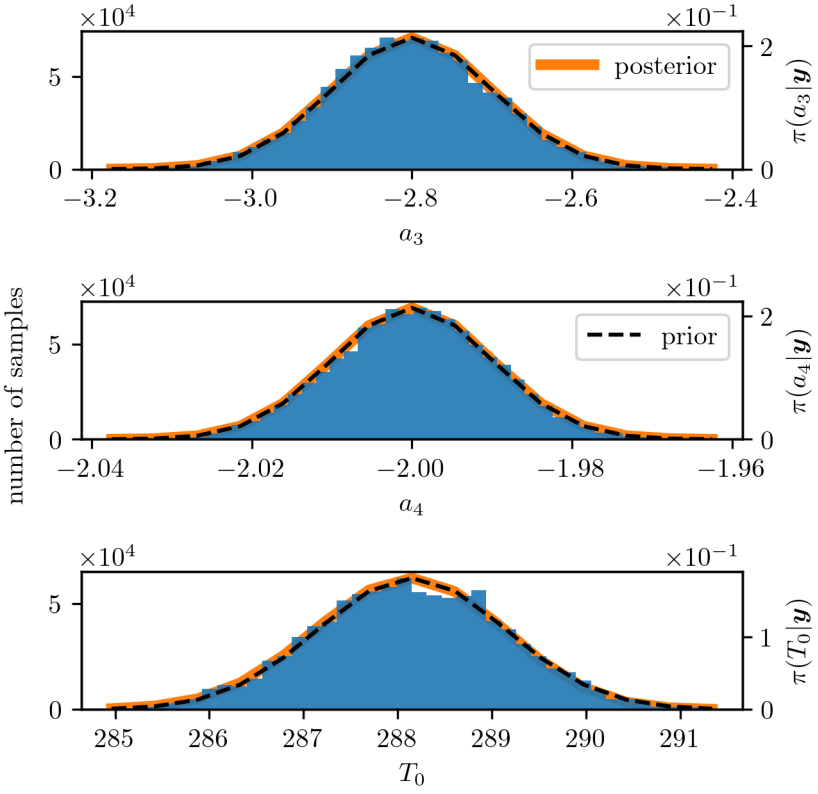

Figure 4.19: We plot the TT approximation of marginal posterior in orange and the samples as a histogram as well as the prior distribution with a dotted line.

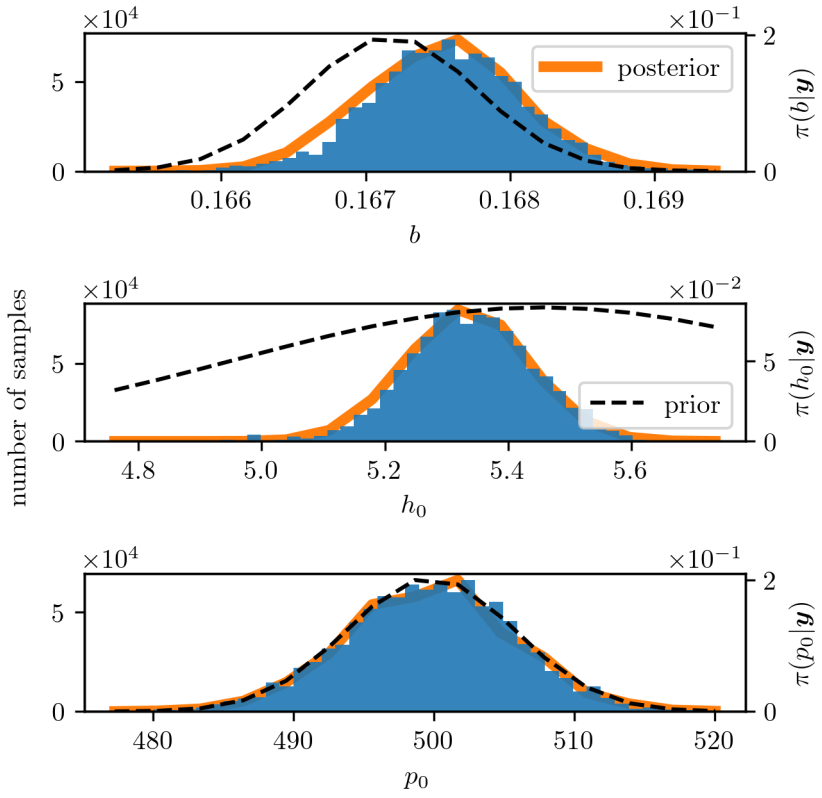

Figure 4.20: We plot the TT approximation of marginal posterior in orange and the samples as a histogram as well as the prior distribution with a dotted line.

Resukts of TT and t-walk overlap say the same for temperatuer we do not gain any more information than priors Priors are posterior For pressure the gain more information grid size refer to figure

Then we can either fit normal distribution to the marginals or draw samples from the output of the t-walk to calculate temperature and pressure profiles according to their respective functions, see Eq. 4.3 and 4.4.

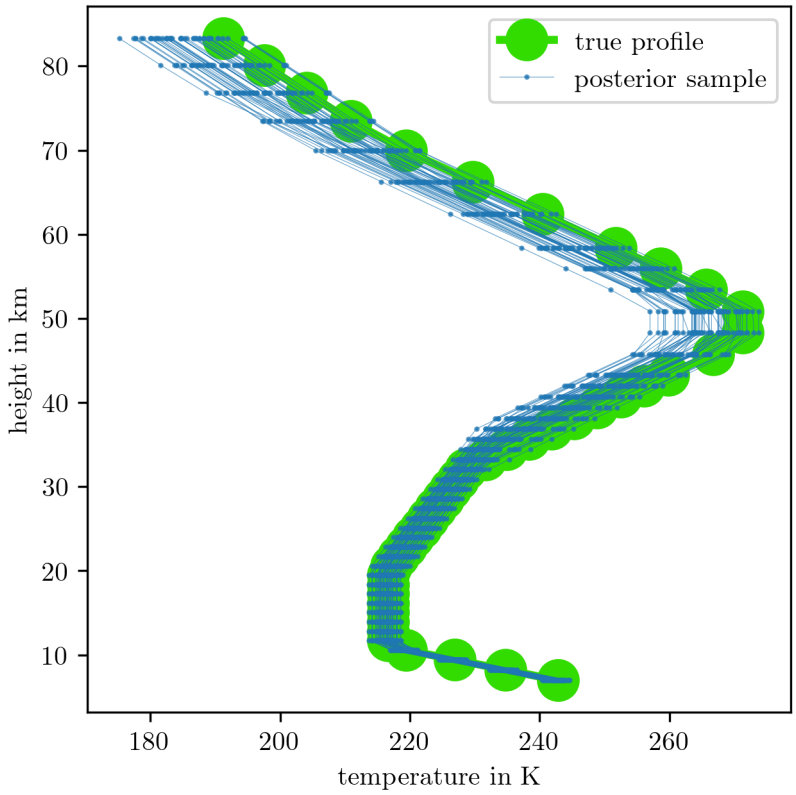

Figure 4.21: We take samples from the posterior distribution, as plotted in Figures 4.16 to 4.19 and plot the corresponding temperature function, see Eq: 4.3.

4.7 Error analysis

In this section we try to estimate errors due to the approximations of functions within the here shown methodology.

Error due to approximation of f and g

When approximating the functions $f(\lambda)$ and $g(\lambda)$ we find that the 3rd order Taylor series of $f(\lambda)$ and a linear approximation of $g(\lambda)$ in log-space gives the smallest error. The maximum absolute error of $f(\lambda)$ is bound by $E_f = \arg\max_{\lambda} f^{(4)}(\lambda_0)/4!\,(\lambda - \lambda_0)^4$, the fourth order Taylor series and corresponds to an relative error of $\approx 20\%$. Since the

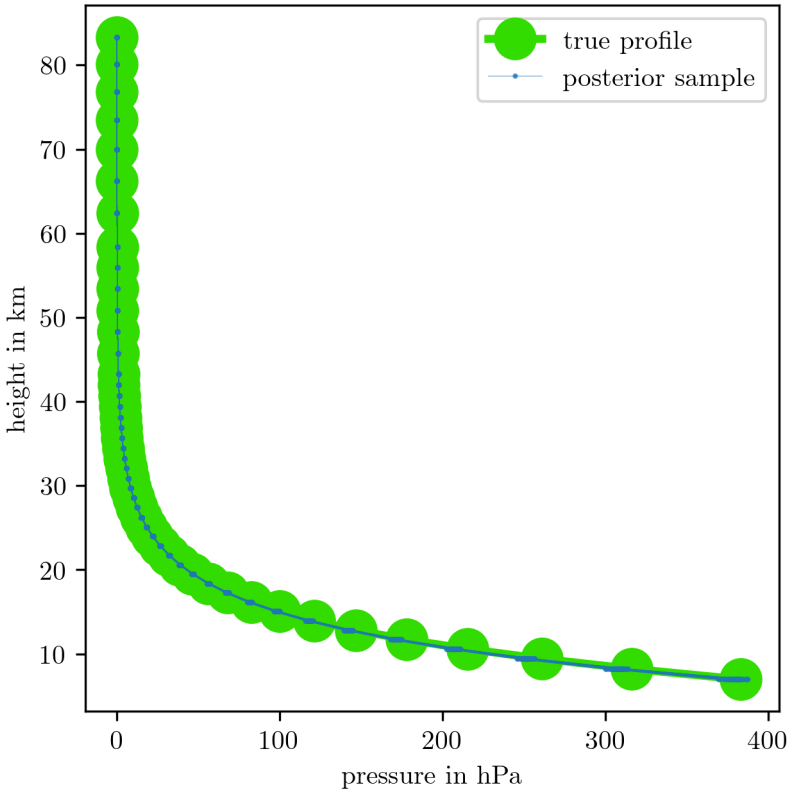

Figure 4.22: We take samples from the posterior distribution, as plotted in Fig. 4.20 and plot the corresponding pressure function, see Eq: 4.4.

maximum absolute error of the approximation $\arg\max_{\lambda} |\tilde{g}(\lambda) - g(\lambda)| \approx 1$ and corresponding to relative error of 0.3%. We neglect the error in g and the maximum relative propagation error $\arg\max_{\lambda,\gamma} 0.5\gamma E_f / \log \pi(\lambda, \gamma | \boldsymbol{y})$ is bound by 5%.

When approximating the marginal posterior the maximum relative propagation error $\arg\max_{\lambda,\gamma} |\tilde{\pi}(\lambda, \gamma | \boldsymbol{y}) - \pi(\lambda, \gamma | \boldsymbol{y})| / |\pi(\lambda, \gamma | \boldsymbol{y})|$ is approximately 100% at $\gamma_{\max}$ and $\lambda_{\max}$, which are the maximum sampled values and lay far away from regions with high probability. We consider negligible because the absolute error in this low probability region is $< 10^{-24} \approx 0$.

Note that one can reduce the maximum errors when approximation $f(\lambda)$ at the mean of $\pi(\lambda, \gamma | \boldsymbol{y})$ as $\pi(\lambda | \boldsymbol{y})$ can be approximated with a skewed distribution, but we don't see noticeable differences in the conditional posterior $\pi(\boldsymbol{x} | \lambda, \gamma, \boldsymbol{y})$ when doing so and consider these errors as tolerable.

ErrorOn the number of sample bins and grid size for the tensor-train approximation

We bin the samples in 25 bins as the realtive error is less than 0.1% which we consider good enough. See Fig.

this is also the gridsize we choose for the TT Of course one coud fo much more precise but it is not worth tit time wise Error analysis and figure and argue TT gridsize

5 Summary and Outlook

5.1 Atmospheric Physics

- Data sensitive informative uninformative, Ozone in higher altitude, pressure , temperature
- SNR v s Pointing accuracy from experience
- include pointing accuracy, weighted mean for pointing accuracy
- nadir geometry for higher altitudes citation

5.2 Methods

- graph Laplacian
- calculating the covariance can be expensive and if that is the case the RTO methods is the preferred choice.
- Through exploratory analysis we found that instead of increasing the ranks optimising the tensors by sweeping over them gives better approximations and is faster, which is crucial in higher dimensions as in section
- TT other bases
- speed gridsize intital ranks
- Machine learning or other methods for affine map
- regularised vs posterior

Appendices

A

Additional MCMC analysis

- integrated autocorrelation time, cite Ulli Wolf [41] pyhton and matlab
- acceptance rate for one D and multi D, cite roberts

B
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

$$\mu \sim \mathcal{N}(0, 1) \tag{B.1a}$$
$$\boldsymbol{x}|\mu \sim \mathcal{N}(\mu\mathbf{1}, \boldsymbol{Q}^{-1}), \tag{B.1b}$$

and apply a Gibbs sampler based on the full conditional distributions

$$\mu^{(k)}|\boldsymbol{x}^{(k)} \sim \mathcal{N}\left(\frac{\mathbf{1}^T\boldsymbol{Q}\boldsymbol{x}^{(k-1)}}{1+\mathbf{1}^T\boldsymbol{Q}\mathbf{1}}, \left(1+\mathbf{1}^T\boldsymbol{Q}\mathbf{1}\right)^{-1}\right) \tag{B.2}$$
$$\boldsymbol{x}^{(k)}|\mu^{(k)} \sim \mathcal{N}(\mu^{(k)}\mathbf{1}, \boldsymbol{Q}^{-1}). \tag{B.3}$$

As illustrated in Figure B.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 B.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.

$$\mu^\star \sim q(\mu^\star|\mu^{(k-1)}) \tag{B.4}$$
$$\boldsymbol{x}^{(k)}|\mu^\star \sim \mathcal{N}(\mu^\star\mathbf{1}, \boldsymbol{Q}^{-1}) \tag{B.5}$$

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.

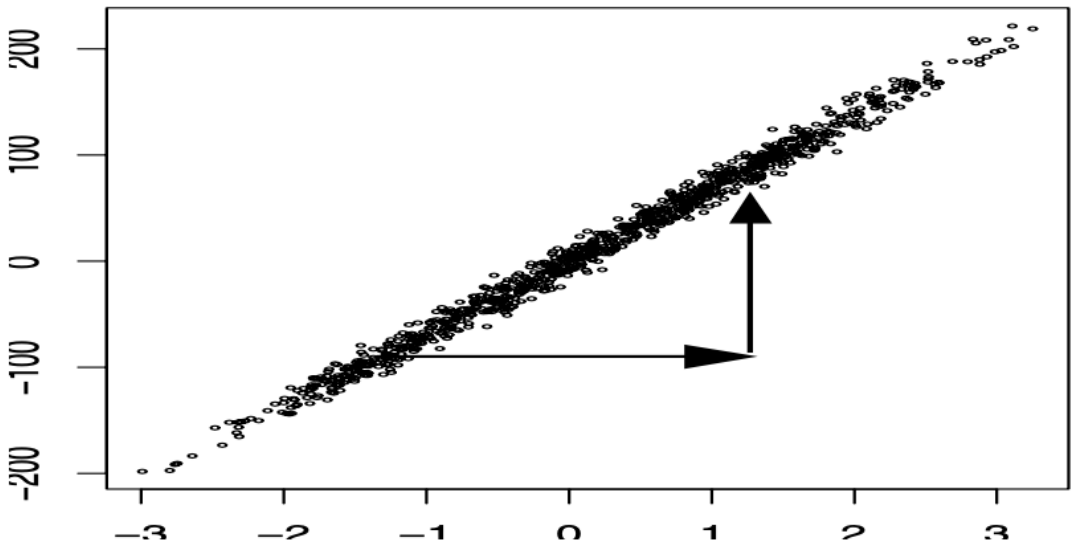

Figure B.1: The figure taken from [10, Figure 4.1 (b)], shows samples from a marginal chain for μ and $\mathbf{1}^T\boldsymbol{Q}\boldsymbol{x}^{(k)}$ over 1000 iterations, based on the hierarchical model in Eq. B.1, with an autoregressive process encoded in $\boldsymbol{Q}$. The algorithm updates μ 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.

C
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 [42] [19] for further reading.

C.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 $\mathbb{P}\left(\bigcup_{j\in\mathbb{N}} A_j\right) = \sum_{j\in\mathbb{N}} \mathbb{P}(A_j)$.

C.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 $A^C \coloneqq A/\Omega \in \mathcal{F}$
- 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.

D
prior modelling

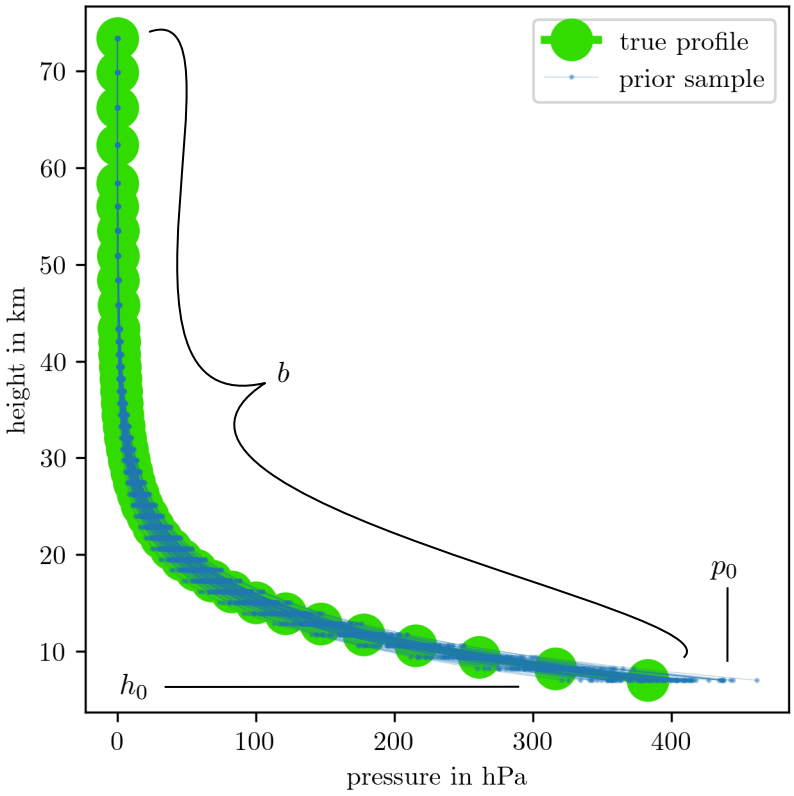

Figure D.1

D.1 t-walk

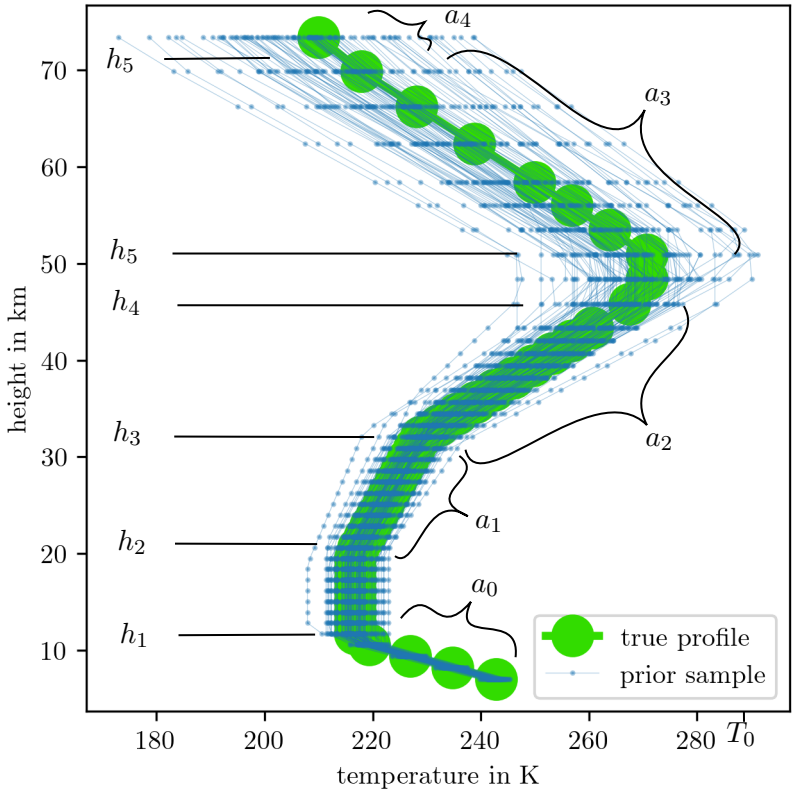
a_4
h_5
a_3
70
60
h_5
50
h_4
40
height in km
h_3
30
a_2
a_1
h_2
20
a_0
h_1
10
true profile
prior sample
180
200
220
240
260
280
T_0
temperature in K

Figure D.2

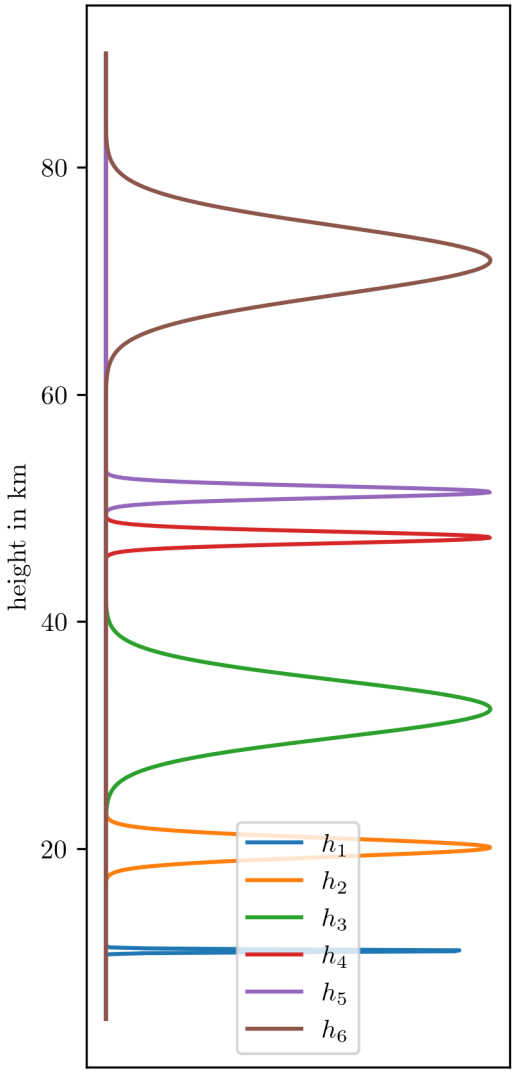

Figure D.3

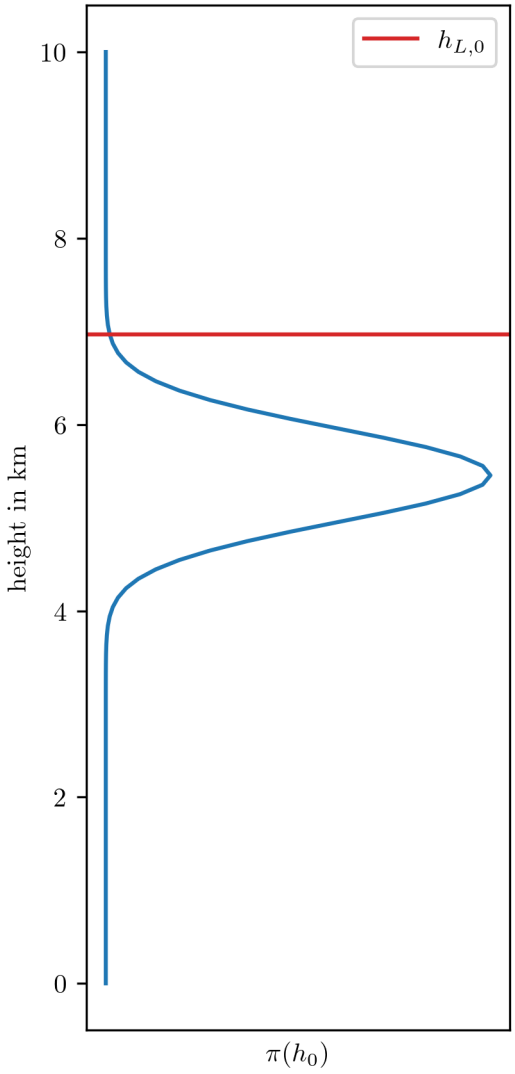

Figure D.4

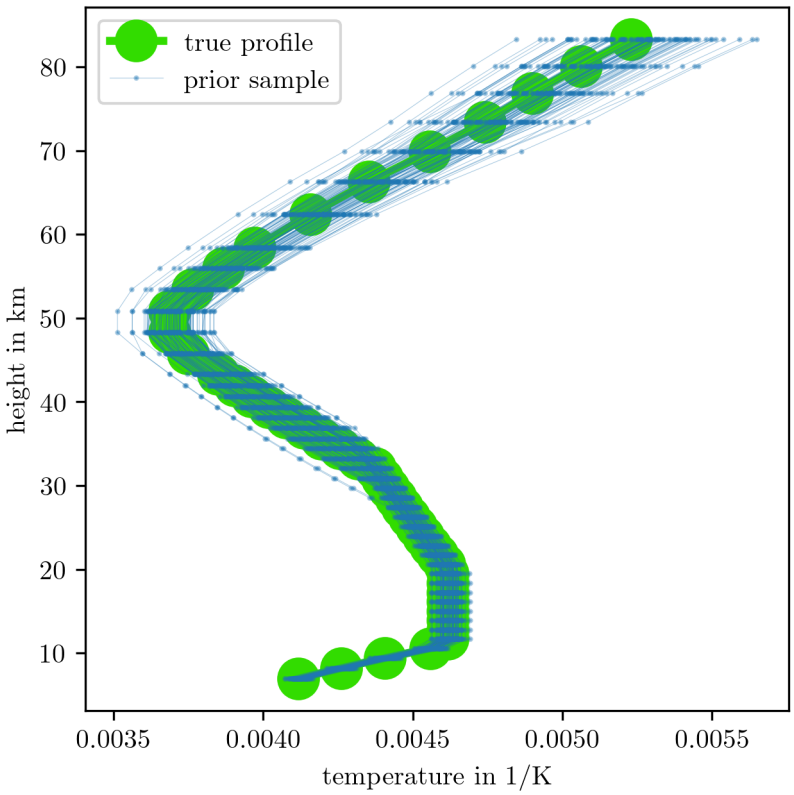
true profile
prior sample
height in km
80
70
60
50
40
30
20
10
0.0035
0.0040
0.0045
0.0050
0.0055
temperature in 1/K

Figure D.5

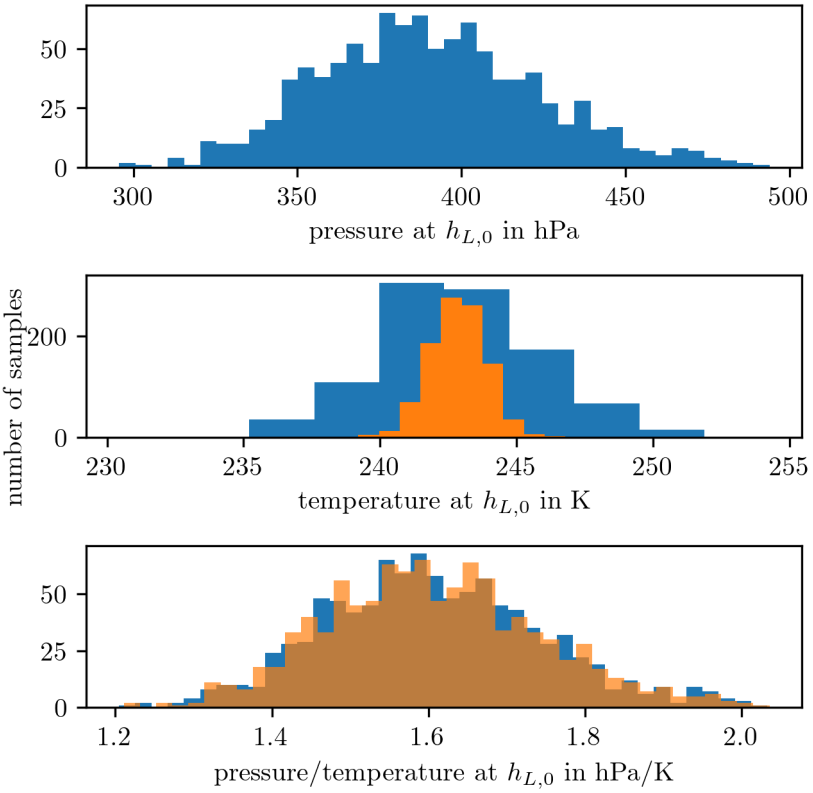
50
25
0
300
350
400
450
500
pressure at $h_{L,0}$ in hPa
number of samples
200
0
230
235
240
245
250
255
temperature at $h_{L,0}$ in K
50
25
0
1.2
1.4
1.6
1.8
2.0
pressure/temperature at $h_{L,0}$ in hPa/K

Figure D.6

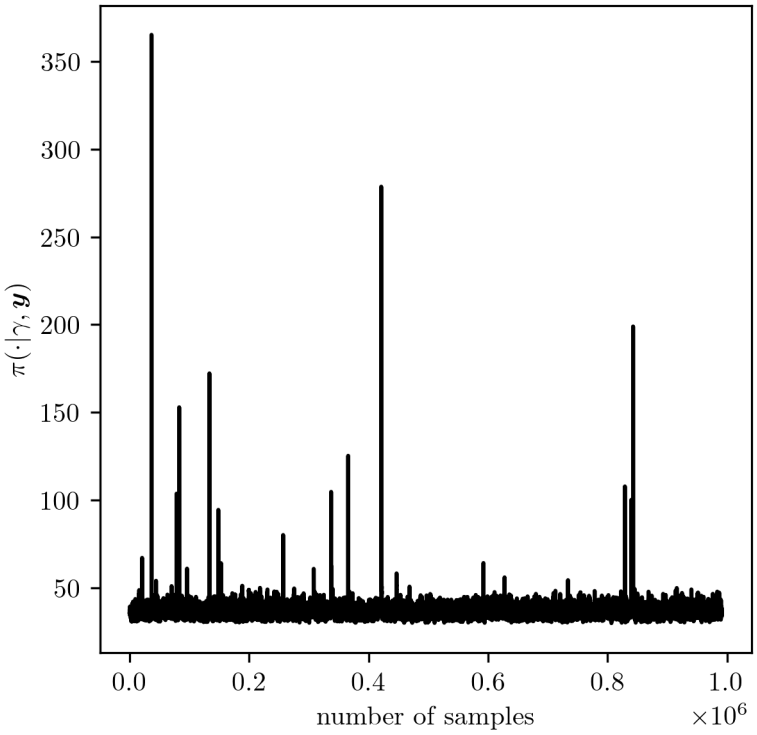
350
300
250
200
150
100
50
$\pi(\cdot|\gamma, \boldsymbol{y})$
0.0
0.2
0.4
0.6
0.8
1.0
number of samples
$\times 10^6$

Figure D.7

References

[1] 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.

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

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

[4] 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.

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

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

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

[8] A. Sokal. "Monte Carlo Methods in Statistical Mechanics: Foundations and New Algorithms". In: *Functional Integration: Basics and Applications*. Ed. by Cecile DeWitt-Morette, Pierre Cartier, and Antoine Folacci. Boston, MA: Springer US, 1997, pp. 131–192.

[9] Ulli Wolff. "Monte Carlo errors with less errors". In: *Computer Physics Communications* 156.2 (2004), pp. 143–153.

[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] 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.

[14] 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.

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

[16] 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.

[17] 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.

[18] 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.

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

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

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

[22] 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.

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

[24] 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.

[25] 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.

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

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

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

[29] Marie Šimečková et al. "Einstein A-coefficients and statistical weights for molecular absorption transitions in the HITRAN database". In: *Journal of Quantitative Spectroscopy and Radiative Transfer* 98.1 (2006), pp. 130–155.

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

[31] Schwartz M. et al. *MLS/Aura Level 2 Ozone (O3) Mixing Ratio V005*. `https://disc.gsfc.nasa.gov/datasets/ML2O3_005/summary?keywords=mls%20o3`. [Online; accessed 25/04/24]. 2020.

[32] *U.S. Standard Atmosphere, 1976*. Washington, D.C.: United States. National Oceanic and Atmospheric Administration, United States Committee on Extension to the Standard Atmosphere, 1976.

[33] Australian National Concurrent Design Facility. *CubeSat Microwave Radiometer Mission to Support Global Ozone Layer Monitoring. Concept Study - Summary Report*. unpublished, internal report. Canberra BC: UNSW Canberra Space, 2023.

[34] H.M. Pickett. "Microwave Limb Sounder THz module on Aura". In: *IEEE Transactions on Geoscience and Remote Sensing* 44.5 (2006), pp. 1122–1130.

[35] Yu-Xiang Wang et al. "Trend Filtering on Graphs". In: *Journal of Machine Learning Research* 17.105 (2016), pp. 1–41. URL: `http://jmlr.org/papers/v17/15-147.html`.

[36] Johnathan M Bardsley et al. "Randomize-then-optimize for sampling and uncertainty quantification in electrical impedance tomography". In: *SIAM/ASA Journal on Uncertainty Quantification* 3.1 (2015), pp. 1136–1158.

[37] Josef Dick, Frances Y. Kuo, and Ian H. Sloan. "High-dimensional integration: The quasi-Monte Carlo way". In: *Acta Numerica* 22 (2013), 133–288.

[38] Per Christian Hansen. *Discrete Inverse Problems: Insight and Algorithms*. Philadelphia: SIAM, 2010.

[39] Ville Satopää et al. “Finding a "Kneedle" in a Haystack: Detecting Knee Points in System Behavior”. In: *2011 31st International Conference on Distributed Computing Systems Workshops*. IEEE. 2011, pp. 166–171.

[40] J. Andrés Christen and Colin Fox. *The t-walk software*. `https://www.cimat.mx/~jac/twalk/`. [Online; accessed 25/11/24].

[41] Ulli Wolff. *UWerr.m Version6*. `https://www.physik.hu-berlin.de/de/com/ALPHAsoft`. [Online; accessed 5/11/23]. 2004.

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