Hierarchical Bayesian Modelling and Posterior Inference Applied to an Atmospheric Limb-Sounder Measuring Ozone

Lennart Golks
Department of Physics

A thesis submitted for the degree of
Doctor of Philosophy

October 2025

Acknowledgements

Abstract

In this thesis, we develop a hierarchical Bayesian model for an atmospheric limb-sounder targeting ozone, as described in [42]. To ensure effective measurements, we briefly assess the informativity of the forward model for different measurement approaches and adapt the data collection accordingly.

Following [19], we utilise the marginal and then conditional scheme to provide posterior distributions of hyper-parameters and ozone profiles, and compare with a Tikhonov regularisation approach. Further, we extend our model and the marginal and then conditional scheme to jointly infer posterior pressure, temperature and ozone.

The main contribution of this work is that we introduce a novel approach to approximate high-dimensional posterior distributions in the tensor-train format [9]. This enables us to generate samples from the target distribution with far fewer function evaluations compared to the t-walk sampling algorithm [7]. Tensor-train methods require a predefined grid and a "normalisation constant" so that function outputs are within computer precision, but once defined, they reduce the function evaluations per independent sample significantly. Another advantage of the tensor-train format is that marginal distributions, useful for characterisation of integrals via quadrature, can be calculated at a low computational cost, without any sampling. To further improve tensor-train methods, we suggest future work should focus on lowering tensor ranks, calculating "normalisation constants" to avoid numerical issues and reducing correlation structures between parameters automatically, all of which we currently have to do by exploratory analysis. Additionally, choosing accurate interpolation schemes between grid points is crucial to improving the effectiveness of the approximation.

Our results show that a hierarchical Bayesian approach, which quantifies posterior mean and variance of the parameter (ozone), provides more information than a regularisation approach at comparable computational time. In regions where the signal strength is low and the data is noise-dominated, we can not recover ozone structures from the ground truth. When including pressure and temperature describing hyper-parameters within our hierarchical Bayesian model, we find a strong correlation between ozone and pressure, whereas the model and data are uninformative about temperature. For future work, we recommend developing a more physically informed parametrised model for ozone within the atmosphere, incorporating atmospheric chemistry and other important processes.

Contents

List of Figures

List of Abbreviations

CDF . . . . . Cumulative Distribution Function

DAG . . . . . Directed Acyclic Graph

HITRAN . . . . . High Resolution Transmission

IACT . . . . . Integrated Autocorrelation Time

IRT . . . . . Inverse Rosenblatt Transform

L . . . . . Linear

MCMC . . . . . Markov Chain Monte-Carlo

MH . . . . . Metropolis-Hastings

MIPAS . . . . . Michelson Interferometer for Passive Atmospheric Sounding

MLS . . . . . Microwave Limb Sounder

MTC . . . . . Marginal and Then Conditional

MVN . . . . . Multivariate Normal

MWG . . . . . Metropolis Within Gibbs

NASA . . . . . National Aeronautics and Space Administration

NL . . . . . Non-Linear

RMS . . . . . Root Mean Square

RTE . . . . . Radiative Transfer Equation

RTO . . . . . Randomise Then Optimise

SIRT . . . . . Squared Inverse Rosenblatt Transform

STD . . . . . Standard Deviation

SVD . . . . . Singular Value Decomposition

TT . . . . . Tensor-Train

VMR . . . . . Volume Mixing Ratio

1 Introduction

Here, we briefly describe the currently used standard to retrieve atmospheric trace gas concentrations from limb-sounding measurements and what motivates us to employ a hierarchical Bayesian framework addressing this inverse problem. We explain how our approach contributes to and improves upon existing methods. Lastly, we provide the reader with the thesis structure.

1.1 Motivation

The only currently operating ozone limb sounder, the Microwave Limb Sounder (MLS) on NASA's Aura satellite, is gradually drifting from its orbit and scheduled to be phased out by 2026 [14]. A group led by Harald Schwefel has proposed an alternative approach to fill this observational gap using a much smaller platform such as a 6U CubeSat (roughly $28\text{cm} \times 15\text{cm} \times 9\text{cm}$) [60]. The proposed system includes a disk-shaped resonator targeting a narrow frequency band and converting the thermal radiation emitted by ozone molecules from the terahertz region to the optical domain [56, 52]. This frequency conversion offers a cost-effective and energy-efficient solution, as it circumvents the need for large, energy-hungry cooling devices that are traditionally required to capture terahertz signals. Instead, signal acquisition in the optical domain can be implemented by using compact, cheap, and low-power photonic technologies.

Currently, the inverse problem to retrieve any trace gas from limb-sounding data is approached by the atmospheric physics community using optimisation and regularisation techniques developed in the 1970s [46, 31]. These methods focus on finding the "best fit to data but not the best fit to parameters" [57]. Instead, we employ a hierarchically structured Bayesian framework to provide a distribution of ozone profiles, which represents

a range of feasible solutions to some given data. This probabilistic approach allows us to determine meaningful estimates and uncertainties of parameters.

1.2 Research Gap and Contribution

As already mentioned, currently the MLS retrieval algorithm [32] is based on the "optimal estimation" method from [46]. This approach provides a point estimate by fitting parameters to some data and iteratively minimising a squared residual norm, penalised against a chosen regularisation. This does not provide comprehensive information about the parameters' underlying correlation structures can lead to unphysical results, e.g. negative ozone concentration values [34], and biased solutions, where the bias is then removed based on empirical decisions [33, 21]. Errors are provided by a local derivative of the forward map at the optimal solution, which is inherently highly sensitive to that specific point in the parameter space. Furthermore, these regularisation methods condition on external point estimates of other parameters, such as temperature or pressure [32].

Naturally, noise (hyper-parameter) is a random process and follows a probability distribution, which we assume to know. According to that noise, as well as some other probability distributions, we can formulate explicit functions over hyper-parameters and parameters (e.g. ozone concentrations).why is this here – not clear to me By incorporating hyper-parameters, e.g. measurement noise and smoothness of the ozone profile, in the modelling and inversion process, we are able to provide errors and a range of feasible paramters (posterior distribution) given some data, instead of one "optimal" solution.what kind of functions - -presently I can't work out what this says This approach is known as hierarchical Bayesian modelling.OH, your trying to say that a hyperparametr controls something about the distribution of the noise, not that the noise is a hyperparameter, which is how this sentence reads. Livesey et al. [32] report "unexpected spectrally correlate noise" on the MLS aura, so here is another real reason why one should include noise in the model.

We utilise the marginal and then conditional (MTC) method [19] to solve this inverse problem and employ a linear-Gaussian hierarchical Bayesian framework. Within seconds we can evaluate the marginal posterior distribution over the hyper-parameters and the conditional posterior distribution over the parameters. this does not clearly say that you treat a nonliner problem later. This is a relatively new method within the Bayesian community, and is the first application to a forward model based on the radiative transfer equation (RTE), to the best of our knowledge. already too many "we". Try something like : the sthe application in this thesis is ... the fist, to the besy of our knowledge.

Moreover, instead of using sampling algorithms to characterise those posterior distributions we approximate posterior distributions directly utilising the tensor-train

(TT) format. likewise. Besides, this sentence is not true. You know about the Approximation and Sampling paper. This enables us to generate independent samples from a TT approximation via the inverse Rosenblatt transform (IRT) with far fewer function evaluations compared to conventional samplers. Moreover, we can calculate marginal distributions of each hyper-parameter and evaluate integrals via quadrature without any sampling.

Since the RTE is weakly non-linear, we approximate the RTE with an affine map, which seems to be another novelty in the field of atmospheric remote sensing.

Additionally, we extend the MTC method to tackle this inverse problem by jointly inferring pressure, temperature and ozone profiles given one set of measurements.

1.3 Thesis Structure

In Chapter 2, we give a brief overview of the key methods used along with references for further reading. Chapter 3 introduces the simplified forward model based on the RTE, and discusses strategies for measuring effectively. Based on this, we simulate noisy data for an idealised limb sounder within a simplified atmosphere. Then, in Chapter **??**, we construct our linear-Gaussian hierarchical Bayesian model and discuss some prior modelling choices. Given the simulated data, we apply the MTC method to provide posterior distributions of our Bayesian framework based on the linearised RTE to then approximate the non-linear forward model with an affine map. We compare a regularisation solution with the posterior distributions of the approximated linear forward model and a ground truth ozone profile. Further, we extend the previously built Bayesian model to include hyper-parameters corresponding to pressure and temperature and touch on some prior modelling choices. Again, using the MTC method we jointly estimate posterior ozone, pressure and temperature profiles and highlight important aspects for improving the effectiveness and stability of TT approximations. Lastly, we summarise and discuss our results and provide an outlook for future work, see Chapter 7.

2

Theoretical and Technical Background

In this Chapter, we introduce and briefly derive the methods used throughout this thesis, and provide references for further reading. We keep it as general as possible, as the expressions specifically tailored towards the forward map will be presented in Chapter **??**. We begin by outlining a general hierarchical Bayesian approach to inverse problems. This is followed by some fundamentals of Markov chain Monte Carlo (MCMC) methods, a specific example of a Metropolis-Hastings algorithm and the randomise then optimise (RTO) method. Further, we present the TT format, which allows to approximate high-dimensional functions and calculate marginal probability distributions cheaply. Lastly, we provide some background information on the Wasserstein distance, affine maps and the Tikhonov regularisation method.

2.1 Hierarchical Bayesian Inference

Assume we observe some data

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

based on a forward model $\boldsymbol{A}(\boldsymbol{x})$, which may be non-linear, an unknown parameter vector $\boldsymbol{x}$ and some additive random noise vector $\boldsymbol{\eta}$. Naturally, due to the noise, which we classify through a hyper-parameter, we deal with a random process. Maybe more clear is to say that the observation process (2.1) is a random process. We incorporate that in our hierarchically ordered modelling through the likelihood function $\pi(\boldsymbol{y}|\boldsymbol{x}, \boldsymbol{\theta})$, which includes all relevant information captured by the model $\boldsymbol{A}(\boldsymbol{x})$, and is defined by the measurement process and the nature of the noise. how does the model "capture information"? Oh, do you mean the observation process? Might pay to not use 'model' for lots of things.

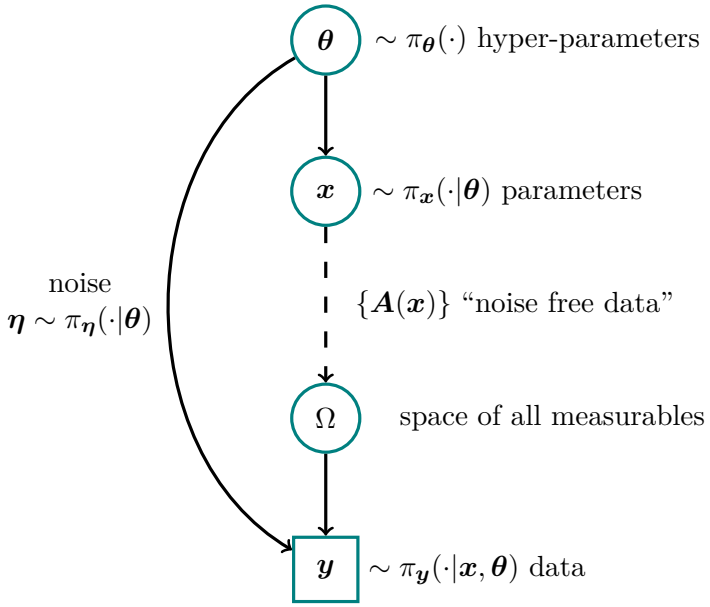


Figure 2.1: The directed acyclic graph (DAG) for an inverse problem visualises statistical dependencies as solid line arrows and deterministic dependencies as dotted arrows. The hyper-parameters $\boldsymbol{\theta}$ are distributed as the hyper-prior distribution $\pi(\boldsymbol{\theta})$. The prior distribution $\pi_{\boldsymbol{x}}(\cdot|\boldsymbol{\theta})$ for the parameter $\boldsymbol{x}$ and the noise $\boldsymbol{\eta} \sim \pi_{\boldsymbol{\eta}}(\cdot|\boldsymbol{\theta})$ are statistically dependent on some of those hyper-parameters. Then a parameter $\boldsymbol{x} \sim \pi_{\boldsymbol{x}}(\cdot|\boldsymbol{\theta})$ is mapped onto the space of all measurables $\Omega = \boldsymbol{A}(\boldsymbol{x})$ deterministically through the forward model. From the space of all measurable noise-free data we observe a data set $\boldsymbol{y} = \boldsymbol{A}(\boldsymbol{x}) + \boldsymbol{\eta}$ with some additive random noise, which determines the likelihood function $\pi(\boldsymbol{y}|\boldsymbol{x}, \boldsymbol{\theta})$.

We read $\pi(\boldsymbol{y}|\boldsymbol{x}, \boldsymbol{\theta})$ as the distribution over $\boldsymbol{y}$ conditioned on the parameter $\boldsymbol{x}$ and the hyper-parameter $\boldsymbol{\theta}$. If we write $\boldsymbol{\eta} \sim \pi_{\boldsymbol{\eta}}(\cdot|\boldsymbol{\theta})$, $\sim$ reads as "is distributed as".

It is intrinsic to hierarchical Bayesian modelling that unknown parameters and hyper-parameters are treated as random variables [29, Chapter 3].

Here $\boldsymbol{\theta}$ may account for multiple hyper-parameters, e.g. describing the distribution $\pi_{\boldsymbol{\eta}}(\cdot|\boldsymbol{\theta})$ over the noise vector $\boldsymbol{\eta}$, and describing physical properties or functional dependencies of $\boldsymbol{x}$, e.g. smoothness, through the prior distribution $\pi(\boldsymbol{x}|\boldsymbol{\theta})$. Consequently $\boldsymbol{\theta}$ is described by the hyper-prior distribution $\pi(\boldsymbol{\theta})$, where $\pi(\boldsymbol{x}, \boldsymbol{\theta}) = \pi(\boldsymbol{x}|\boldsymbol{\theta})\pi(\boldsymbol{\theta})$. Choosing these prior distributions is ultimately a modeller's choice and is crucial, as it shall be as uninformative as possible for regions in hyper-parameter and parameter space where the data is informative. If the data is uninformative the prior distributions are rather informative and represent a range of (physically) feasible hyper-parameters and parameters.

not affect the posterior distribution what, of course the prior affects the posterior. What are you trying to say? According to Bayes' theorem the posterior distribution is given as

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

with finite and non-zero $\pi(\boldsymbol{y})$. $\pi(\boldsymbol{x}, \boldsymbol{\theta}|\boldsymbol{y})$ reads as the distribution of $\boldsymbol{x}$ and $\boldsymbol{\theta}$ given (conditioned on) the data $\boldsymbol{y}$. two sentences, talk anbout posterior and then model

Hence, the posterior is a distribution over the parameters $\boldsymbol{x}$ and the hyper-parameter $\boldsymbol{\theta}$. We can visualise this hierarchically-ordered correlation structure between parameters as well as how distributions progress through a measurement process, using a directed acyclic graph (DAG) as in Figure 2.1. conditional dependencies and sqaure box is observation

Note that here we include the hyper-parameters within the posterior distribution, which is the key idea of hierarchical Bayesian modelling. "the" key idea? Maybe say that it is intrinsic to Bayes models, or something. You could quote Kaipio and Somersalo that all unknowns are treated as random variables. [29, Chapter 3]

Usually, the objective is to calculate the expectation of a function $h(\boldsymbol{x})$, which is defined as

$$\mathrm{E}_{\boldsymbol{x},\boldsymbol{\theta}|\boldsymbol{y}}[h(\boldsymbol{x})] = \underbrace{\int\int h(\boldsymbol{x})\,\pi(\boldsymbol{x},\boldsymbol{\theta}|\boldsymbol{y})\;\mathrm{d}\boldsymbol{x}\;\mathrm{d}\boldsymbol{\theta}}_{\bar{h}}\,. \tag{2.3}$$

If that is a high-dimensional integral and computationally not feasible to solve, we approximate

$$\mathrm{E}_{\boldsymbol{x},\boldsymbol{\theta}|\boldsymbol{y}}[h(\boldsymbol{x})] \approx \underbrace{\frac{1}{N}\sum_{k=1}^{N} h(\boldsymbol{x}^{(k)})}_{\bar{h}_N}\,, \tag{2.4}$$

with an the unbiased sample-based Monte-Carlo estimate [44] for large enough N (law of large numbers [35, Chapter 17]). Here, the posterior samples $\{\boldsymbol{x}^{(k)},\boldsymbol{\theta}^{(k)}\} \sim \pi_{\boldsymbol{x},\boldsymbol{\theta}}(\cdot|\boldsymbol{y})$, for $k = 1,\dots,N$, form a sample set $\mathcal{M} = \{(\boldsymbol{x},\boldsymbol{\theta})^{(1)},\dots,(\boldsymbol{x},\boldsymbol{\theta})^{(N)}\}$. The central limit theorem states that the sample mean $\bar{h}_N^{(i)}$ of independent sample sets $\mathcal{M}^{(i)} \sim \pi(\boldsymbol{x},\boldsymbol{\theta}|\boldsymbol{y})$, for $i = 1,\dots,n$ from a distribution, converges to be normally distributed, so that

$$\sqrt{n}(\bar{h}_N^{(i)} - \bar{h}) \xrightarrow{\mathcal{D}} \mathcal{N}(0,\sigma^2)[22], \tag{2.5}$$

and if $\sigma^2 < \infty$ the Monte-Carlo error $\bar{h}_N^{(i)} - \bar{h}$ is bounded. In practice, we approximate the estimation error from a sample set $\mathcal{M}^{(i)}$ as

$$(\sigma^{(i)})^2 = \mathrm{Var}(\mathrm{E}_{\boldsymbol{x},\boldsymbol{\theta}|\boldsymbol{y}}[h(\boldsymbol{x})]) \approx \frac{\mathrm{Var}(h(\boldsymbol{x}))}{N}\underbrace{\left(1 + 2\sum_{t=1}^{W}\frac{\Gamma(t)}{\Gamma(0)}\right)}_{:=\tau_{\mathrm{int}}} = \mathrm{Var}(h(\boldsymbol{x}))\frac{\tau_{\mathrm{int}}}{N}\,, \tag{2.6}$$

where we have to take into account that the samples generated by any system or algorithm are correlated. We define the integrated autocorrelation time (IACT) τ_{int} as in [19], which is twice the value of the IACT in [66, pp. 103-105]. Here the autocorrelation coefficient $\Gamma(t) \propto \exp\{-|t|/\tau\} \longrightarrow 0$ for $t \to \infty$ at lag t decays exponentially and $\Gamma(0) = \mathrm{Var}(h(\boldsymbol{x}))$. Choosing the summation window W is crucial because it has to be large compared to the decay time τ, but for too large t the autocorrelation coefficient $\Gamma(t)$ is noise-dominated.

U. Wolff [64] (and the Python implementation by D. Hesse [28]) provide a way to not only calculate the IACT safely but also to quantify the errors of the estimated IACT.

The IACT provides a good estimate of the number of steps the sampling algorithm needs to take to produce one independent sample. According The IACT, we define the effective sample size as τ_{int}/N. We point out that for uncorrelated samples $\tau_{\text{int}} = 1$ the error $(\sigma^{(i)})^2$ is a typical Monte-Carlo estimate. See Appendix A.2 and [55, 64, 66] for a more detailed derivation.

Ergodicity

In this section we present the methods that draw samples from the marginal posterior $\mathcal{M} = \{\boldsymbol{\theta}^{(1)}, \dots, \boldsymbol{\theta}^{(k)}, \dots, \boldsymbol{\theta}^{(N)}\} \sim \pi(\boldsymbol{\theta}|\boldsymbol{y})$ this reads as if M is the marginal posterior. Rewrite. as well as the RTO method to draw samples from a normally distributed full conditional posterior $\pi(\boldsymbol{x}|\boldsymbol{\theta}, \boldsymbol{y})$. what is the "full posterior" as opposed to the "posterior"? Here, $\mathcal{M}$ denotes a Markov chain, where each new sample $\boldsymbol{\theta}^{(k)}$ is only affected by the previous one $\boldsymbol{\theta}^{(k-1)}$. MCMC methods generate such a chain $\mathcal{M}$ using random (Monte-Carlo) proposals $(\boldsymbol{x}, \boldsymbol{\theta})^{(k)} \sim q(\cdot|\boldsymbol{\theta}^{(k-1)})$ This is defining "Markov". . Be more explicit. according to a proposal distribution conditioned on the previous sample (Markov), where ergodicity of the chain $\mathcal{M}$ is a sufficient criterion for using sample-based estimates [57, 44]. you are a little unclear about a chain of random variables, and an instance of such a chain.

The ergodicity theorem in [57] 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, the chain can reach any state from any other state (irreducibility), is not stuck in periodic cycles (aperiodicity), and satisfies the detailed balance condition [57] (reversibility). Then the samples in that chain $\mathcal{M} \sim \pi(\boldsymbol{\theta}|\boldsymbol{y})$ are samples from the desired target distribution. In practice, one can inspect the trace $\pi(\boldsymbol{\theta}^{(k)}|\boldsymbol{y})$ for $k = 1, \dots, N$ and visually assess convergence and mixing properties of the chain to evaluate ergodicity. converge to, so after 'burn in' The sampling methods used in this thesis possess proven ergodic properties, and we therefore refer the reader to the corresponding literature for further details. this is a bit loose. really, one evaluates if the behaviour is consistent with ergodicity.

If the Markov chain over the marginal posterior $\pi(\boldsymbol{\theta}|\boldsymbol{y})$ is ergodic reference Jun Liu, or something., and the full conditional samples $\boldsymbol{x}^{(k)} \sim \pi(\boldsymbol{x}|\boldsymbol{\theta}^{(k)}, \boldsymbol{y})$ are drawn independently, as e.g. in Sec. 6.3.1, then the resulting joint chain $\{(\boldsymbol{x}, \boldsymbol{\theta})^{(1)}, \dots, (\boldsymbol{x}, \boldsymbol{\theta})^{(N)}\} \sim \pi(\boldsymbol{x}, \boldsymbol{\theta}|\boldsymbol{y}) = \pi(\boldsymbol{x}|\boldsymbol{\theta}, \boldsymbol{y})\pi(\boldsymbol{\theta}|\boldsymbol{y})$ is also ergodic for the posterior [61]. strictly, ergodic for [theta|y] – ergodicity is with respect to a particular distribution.

2.1.1 Marginal and then Conditional Method

Generating a representative sample set quickly from the posterior distribution often presents a significant challenge. This is mainly this word is misplaced (for English). Better is Quickly generating (it's an adverb) due to the strong correlations that usually exist between the parameters and hyper-parameters, as discussed by Rue and Held in [48] and illustrated in Appendix A.1. If $\boldsymbol{x}$ cannot be parametrised directly in terms of the hyper-parameters $\boldsymbol{\theta}$, so that $\boldsymbol{x}(\boldsymbol{\theta})$ is function of $\boldsymbol{\theta}$, it is beneficial to factorise the posterior distribution as this is unclear. Of course x(theta) is a function of theta. What are you trying to say?

$$\pi(\boldsymbol{x}, \boldsymbol{\theta}|\boldsymbol{y}) = \pi(\boldsymbol{x}|\boldsymbol{\theta}, \boldsymbol{y})\, \pi(\boldsymbol{\theta}|\boldsymbol{y}), \tag{2.7}$$

into the full conditional posterior $\pi(\boldsymbol{x}|\boldsymbol{\theta}, \boldsymbol{y})$ over the latent field $\boldsymbol{x}$ and the marginal posterior , that can be calculated by,

$$\pi(\boldsymbol{\theta}|\boldsymbol{y}) = \frac{\pi(\boldsymbol{y}|\boldsymbol{x}, \boldsymbol{\theta})\pi(\boldsymbol{x}|\boldsymbol{\theta})\pi(\boldsymbol{\theta})}{\pi(\boldsymbol{x}|\boldsymbol{\theta}, \boldsymbol{y})\pi(\boldsymbol{y})} \propto \frac{\pi(\boldsymbol{y}|\boldsymbol{x}, \boldsymbol{\theta})\pi(\boldsymbol{x}|\boldsymbol{\theta})\pi(\boldsymbol{\theta})}{\pi(\boldsymbol{x}|\boldsymbol{\theta}, \boldsymbol{y})}, \tag{2.8}$$

over the hyper-parameter $\boldsymbol{\theta}$ (see [19, Lemma 2]).

This approach, known as the MTC method, is particularly advantageous when $\boldsymbol{x} \in \mathbb{R}^n$ is high-dimensional, while $\boldsymbol{\theta} \in \mathbb{R}^{n_{\boldsymbol{\theta}}}$ is low-dimensional, so that $n_{\boldsymbol{\theta}} \ll n$ and the evaluation of $\pi(\boldsymbol{\theta}|\boldsymbol{y})$ is relatively cheap. horrible notation, try better. Applying the law of total expectation [6], Eq. (2.3) becomes

$$\begin{aligned}
\mathbb{E}_{\boldsymbol{x},\boldsymbol{\theta}|\boldsymbol{y}}[h(\boldsymbol{x})] &= \int\int h(\boldsymbol{x})\pi(\boldsymbol{x}|\boldsymbol{\theta}, \boldsymbol{y})\ \mathrm{d}\boldsymbol{x}\, \pi(\boldsymbol{\theta}|\boldsymbol{y})\ \mathrm{d}\boldsymbol{\theta} && (2.9)\\
&= \int \mathbb{E}_{\boldsymbol{x}|\boldsymbol{\theta},\boldsymbol{y}}\left[h(\boldsymbol{x})\right]\, \pi(\boldsymbol{\theta}|\boldsymbol{y})\ \mathrm{d}\boldsymbol{\theta} && (2.10)\\
&= \mathbb{E}_{\boldsymbol{\theta}|\boldsymbol{y}}\left[\mathbb{E}_{\boldsymbol{x}|\boldsymbol{\theta},\boldsymbol{y}}[h(\boldsymbol{x})]\right], && (2.11)
\end{aligned}$$

where, in the case of a linear-Gaussian hierarchical Bayesian model, both the marginal distribution $\pi(\boldsymbol{\theta}|\boldsymbol{y})$ separate sentence (You have a tendency to let sentences wander to different ideas. Split into single topics.) and the inner expectation $\mathbb{E}_{\boldsymbol{x}|\boldsymbol{\theta},\boldsymbol{y}}\left[h(\boldsymbol{x})\right]$ are well defined (see next subsection). If the integral in Eq. 2.10 is expensive to calculate, we use sample-based methods to produce a Markov chain $\{(\boldsymbol{x}, \boldsymbol{\theta})^{(1)}, \ldots, (\boldsymbol{x}, \boldsymbol{\theta})^{(N)}\} \sim \pi(\boldsymbol{x}, \boldsymbol{\theta}|\boldsymbol{y})$ and sample from $\pi(\boldsymbol{\theta}|\boldsymbol{y})$ first and then draw samples from the full conditional posterior $\pi(\boldsymbol{x}|\boldsymbol{\theta}, \boldsymbol{y})$, e.g via the RTO method (see Sec. 6.3.1).

Linear-Gaussian hierarchical Bayesian model

In case of normally distributed noise $\boldsymbol{\eta} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\Sigma}(\boldsymbol{\theta}))$ with zero mean and covariance $\boldsymbol{\Sigma}(\boldsymbol{\theta})$, a linear forward model matrix $\boldsymbol{A}$, and data Eq (2.1) simplifies to

$$\boldsymbol{y} = \boldsymbol{A}\boldsymbol{x} + \boldsymbol{\eta}\,. \tag{2.12}$$

Then we can derive the marginal and full conditional posterior distribution explicitly. We define our hierarchical linear-Gaussian Bayesian model as

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

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

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

is this indented? with a Gaussian likelihood function $\pi(\boldsymbol{y}|\boldsymbol{x}, \boldsymbol{\theta})$, a normally distributed prior $\pi(\boldsymbol{x}|\boldsymbol{\theta})$, with prior mean $\boldsymbol{\mu}$ and prior precision $\boldsymbol{Q}(\boldsymbol{\theta})$, and a hyper-prior distribution $\pi(\boldsymbol{\theta})$. To derive the marginal posterior and the full conditional posterior distribution, we consider the joint multivariate Gaussian distribution

$$\begin{pmatrix} \boldsymbol{x} \\ \boldsymbol{y} \end{pmatrix} \sim \mathcal{N}\left[\begin{pmatrix} \boldsymbol{\mu} \\ \boldsymbol{A}\boldsymbol{\mu} \end{pmatrix}, \begin{pmatrix} \boldsymbol{Q}(\boldsymbol{\theta}) + \boldsymbol{A}^T\boldsymbol{\Sigma}(\boldsymbol{\theta})^{-1}\boldsymbol{A} & -\boldsymbol{A}^T\boldsymbol{\Sigma}(\boldsymbol{\theta})^{-1} \\ \boldsymbol{\Sigma}(\boldsymbol{\theta})^{-1}\boldsymbol{A} & \boldsymbol{\Sigma}(\boldsymbol{\theta})^{-1} \end{pmatrix}^{-1} \right], \tag{2.14}$$

where we provide the joint precision matrix as in [54] (see also [48, 19]). Immediately, we formulate the full conditional posterior distribution as

$$\boldsymbol{x}|\boldsymbol{\theta}, \boldsymbol{y} \sim \mathcal{N}\big(\underbrace{\boldsymbol{\mu} + (\boldsymbol{Q}(\boldsymbol{\theta}) + \boldsymbol{A}^T\boldsymbol{\Sigma}(\boldsymbol{\theta})^{-1}\boldsymbol{A})^{-1}(\boldsymbol{y} - \boldsymbol{A}\boldsymbol{\mu})}_{\boldsymbol{\mu}_{\boldsymbol{x}|\boldsymbol{\theta},\boldsymbol{y}}}, \underbrace{(\boldsymbol{Q}(\boldsymbol{\theta}) + \boldsymbol{A}^T\boldsymbol{\Sigma}(\boldsymbol{\theta})^{-1}\boldsymbol{A})^{-1}}_{\boldsymbol{\Sigma}_{\boldsymbol{x}|\boldsymbol{\theta},\boldsymbol{y}}} \big)\,. \tag{2.15}$$

Then the marginal posterior distribution over the hyper-parameters can be derived as in Eq. 2.8, where, as noted by Fox et al. [19], the parameter $\boldsymbol{x}$ cancels and we arrive at

$$\begin{aligned} \pi(\boldsymbol{\theta}|\boldsymbol{y}) \propto & \sqrt{\frac{\det\left(\boldsymbol{\Sigma}(\boldsymbol{\theta})^{-1}\right)\det\left(\boldsymbol{Q}(\boldsymbol{\theta})\right)}{\det\left(\boldsymbol{Q}(\boldsymbol{\theta}) + \boldsymbol{A}^T\boldsymbol{\Sigma}(\boldsymbol{\theta})^{-1}\boldsymbol{A}\right)}} \exp\Bigg\{ -\frac{1}{2}(\boldsymbol{y} - \boldsymbol{A}\boldsymbol{\mu})^T \\ & \left[\boldsymbol{\Sigma}(\boldsymbol{\theta})^{-1} - \boldsymbol{\Sigma}(\boldsymbol{\theta})^{-1}\boldsymbol{A}(\boldsymbol{Q}(\boldsymbol{\theta}) + \boldsymbol{A}^T\boldsymbol{\Sigma}(\boldsymbol{\theta})^{-1}\boldsymbol{A})^{-1}\boldsymbol{A}^T\boldsymbol{\Sigma}(\boldsymbol{\theta})^{-1}\right](\boldsymbol{y} - \boldsymbol{A}\boldsymbol{\mu}) \Bigg\} \pi(\boldsymbol{\theta})\,. \end{aligned} \tag{2.16}$$

Having the marginal posterior distribution $\pi(\boldsymbol{\theta}|\boldsymbol{y})$ (independent of $\boldsymbol{x}$) available breaks up the correlation structure between $\boldsymbol{x}$ and $\boldsymbol{\theta}$ (see Appendix A.1), and makes the MTC approach very efficient [19]. put this before the (), otherwise it reads very strangely Within this scheme, we evaluate the marginal posterior first and then either condition

on hyper-parameters to draw full conditional posterior samples $\boldsymbol{x} \sim \pi(\boldsymbol{x}|\boldsymbol{\theta}, \boldsymbol{y})$ (see Sec. 6.3.1) or evaluate the mean

$$\mu_{\boldsymbol{x}|\boldsymbol{y}} = \int \boldsymbol{\mu}_{\boldsymbol{x}|\boldsymbol{\theta},\boldsymbol{y}} \pi(\boldsymbol{\theta}|\boldsymbol{y}) \, \mathrm{d}\boldsymbol{\theta} \tag{2.17}$$

and covariance matrix

$$\Sigma_{\boldsymbol{x}|\boldsymbol{y}} = \int \boldsymbol{\Sigma}_{\boldsymbol{x}|\boldsymbol{\theta},\boldsymbol{y}} \pi(\boldsymbol{\theta}|\boldsymbol{y}) \, \mathrm{d}\boldsymbol{\theta} \tag{2.18}$$

of the full posterior what is the "full posterior" as opposed to the "posterior"? $\pi(\boldsymbol{x}|\boldsymbol{y})$ by some quadrature rule.

2.2 Numerical Approximation – Tensor-Train (TT)

Instead of relying on sampling-based methods to explore a target distribution $\pi(\boldsymbol{x})$, we can approximate this distribution on a d-dimensional grid using a TT approximation $\tilde{\pi}(\boldsymbol{x}) \approx \pi(\boldsymbol{x})$, where $\boldsymbol{x} \in \mathbb{R}^d$, with far fewer function evaluation compared to conventional sampling methods. In the following, we describe how to compute marginal distributions from a probability density approximated in TT (tensor train) format and how to generate samples using the inverse Rosenblatt transform (IRT), following the notation and procedure introduced by Cui et al. [9].

I'm surprised that you never cite the Approximation an Sampling paper that introduced the idea of TT approximation of PDFs. Oh, you do, but the author list is abbreviated. Don't abbreviate the author list fro a thesis. As in [9], we can define the parameter space as the product space $\mathcal{X} = \mathcal{X}_1 \times \mathcal{X}_2 \times \cdots \times \mathcal{X}_d$ with $x_k \in \mathcal{X}_k \subseteq \mathbb{R}$. The marginal density function for the k-th component is then given by

$$f_{X_k}(x_k) = \frac{1}{z} \int_{\mathcal{X}_1} \cdots \int_{\mathcal{X}_{k-1}} \int_{\mathcal{X}_{k+1}} \cdots \int_{\mathcal{X}_d} \boldsymbol{\lambda}(\boldsymbol{x}) \, \pi(\boldsymbol{x}) \, \mathrm{d}x_1 \cdots \mathrm{d}x_{k-1} \, \mathrm{d}x_{k+1} \cdots \mathrm{d}x_d, \tag{2.19}$$

where we integrate over all dimensions except the k-th, and z is a normalisation constant. Here, we introduce a weight function $\boldsymbol{\lambda}(x)$, which can be useful for quadrature rules [10], which [9] refers as a "product-form Lebesgue-measurable weighting function" and define it as

$$\boldsymbol{\lambda}(\mathcal{X}) = \prod_{i=1}^{d} \boldsymbol{\lambda}_i(\mathcal{X}_i), \quad \text{where} \quad \boldsymbol{\lambda}_i(\mathcal{X}_i) = \int_{\mathcal{X}_i} \boldsymbol{\lambda}_i(x_i) \, \mathrm{d}x_i. \tag{2.20}$$

In the TT format, the integral in Eq. 2.19 for the marginal probability can be computed at a low computational cost, as $\pi(\boldsymbol{x})$ is approximated by

$$\tilde{\pi}(\boldsymbol{x}) = \tilde{\pi}_1(x_1)\tilde{\pi}_2(x_2) \cdots \tilde{\pi}_d(x_d),$$

which is a sequence of matrix multiplications with $\tilde{\pi}_k(x_k) \in \mathbb{R}^{r_{k-1} \times r_k}$ for a fixed point $\boldsymbol{x} = (x_1, \dots, x_d)$ on a predefined d-dimensional discrete univariate grid over the parameter space $\mathcal{X}$. We call $\tilde{\pi}_k \in \mathbb{R}^{r_{k-1} \times n \times r_k}$ a TT core with ranks r_{k-1} and r_k, where the outer ranks are $r_0 = r_d = 1$, representing each dimension on n grid points and connecting to neighbouring dimensions through its ranks. more sentence creep. You should have defined the univariate grids, earlier. This enables us to approximate $\pi(\mathcal{X}) \approx \tilde{\pi}_1 \tilde{\pi}_2 \cdots \tilde{\pi}_d$ over a discrete parameter space $\mathcal{X}$ using $2nr + (d-2)nr^2$ evaluation points for fixed ranks $r = r_{k-1} = r_k$, as illustrated in Figure 2.2, instead of n^d function evaluation. Consequently, the marginal target distribution

$$f_{X_k}(x_k) \approx \frac{1}{z} \Big| \left(\int_{\mathcal{X}_1} \lambda_1(x_1)\tilde{\pi}_1(x_1) \, \mathrm{d}x_1 \right) \cdots \left(\int_{\mathcal{X}_{k-1}} \lambda_{k-1}(x_{k-1})\tilde{\pi}_{k-1}(x_{k-1}) \, \mathrm{d}x_{k-1} \right) \lambda_k(x_k)\tilde{\pi}_k(x_k) \left(\int_{\mathcal{X}_{k+1}} \lambda_{k+1}(x_{k+1})\tilde{\pi}_{k+1}(x_{k+1}) \, \mathrm{d}x_{k+1} \right) \cdots \left(\int_{\mathcal{X}_d} \lambda_d(x_d)\tilde{\pi}_d(x_d) \, \mathrm{d}x_d \right) \Big| \tag{2.21}$$

is computed by integrating over all TT cores except the k-th core π_k, as in [13], and normalised by the constant z [9].

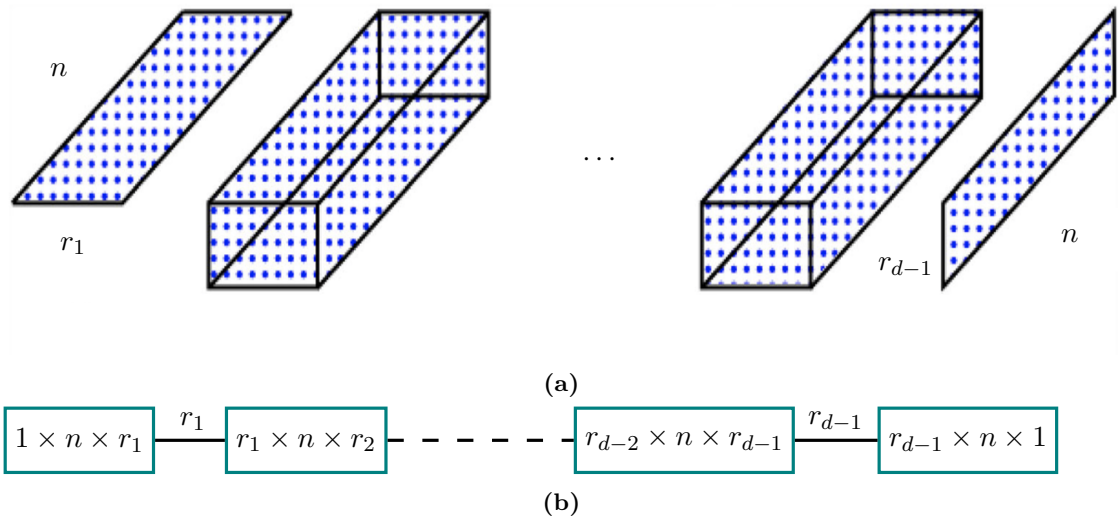


Figure 2.2: Here, we visualise the TT cores as a train of two- and three-dimensional matrices. Each core has a length n, corresponding to the number of grid points in each dimension, and the cores are connected through ranks r_k. More specifically, a core $\tilde{\pi}_k$ has dimensions $r_{k-1} \times n \times r_k$, with outer ranks $r_0 = r_d = 1$. Using the TT format enables us to represent a d-dimensional grid with only $2nr + (d-2)nr^2$ evaluation points instead of n^d grid points. Figure (a) is adapted from [20].

In practice, TT approximations may suffer from numerical instability, in particular because it is not advantageous yet to approximate the target function $\pi(\boldsymbol{x})$ in e.g. the logarithmic space. Hence, Cui et al. [9] approximate the square root of the probability

density Thsi sentence needs rewriting - -it's a jumble of ideas. Hence implies an implication – is it that or just one possible resolution?

$$\sqrt{\pi(\boldsymbol{x})} \approx \tilde{g}(\boldsymbol{x}) = \boldsymbol{G}_1(x_1), \ldots, \boldsymbol{G}_k(x_k), \ldots, \boldsymbol{G}_d(x_d) \text{ [9, Eq. 18]}, \tag{2.22}$$

which ensures positivity. Here, each TT core is given by

$$G_k^{(\alpha_{k-1},\alpha_k)}(x_k) = \sum_{i=1}^{n_k} \phi_k^{(i)}(x_k) \boldsymbol{A}_k[\alpha_{k-1}, i, \alpha_k], \quad k = 1, \ldots, d, \text{ [9, Eq. 21]}, \tag{2.23}$$

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 unnormalised density is written as: put the reference in the text. Immediately prior is probably best.

$$\pi(\boldsymbol{x}) \approx \xi + \tilde{g}(\boldsymbol{x})^2 \text{ [9, Eq. 19]}, \tag{2.24}$$

where ξ is a positive constant added according to the ratio of the Lebesgue weighted L2-norm error and the Lebesgue weighting (see Eq. 2.20) such that

$$0 \leq \xi \leq \frac{1}{\lambda(\mathcal{X})} \|\tilde{g} - \sqrt{\pi}\|^2_{L^2_\lambda(\mathcal{X})} \text{ [9, Eq. 35]}. \tag{2.25}$$

This leads to the normalised target function

$$f_X(\boldsymbol{x}) \approx \frac{1}{z} \left(\lambda(\boldsymbol{x})\xi + \lambda(\boldsymbol{x})\tilde{g}(\boldsymbol{x})^2 \right) \text{ [9, Eq. 19]}, \tag{2.26}$$

with the normalisation constant $z = \int_{\mathcal{X}} f_X(\boldsymbol{x}) \, \mathrm{d}\boldsymbol{x}$. Given the tensor train approximation of $\sqrt{\pi}$, the marginal function $f_{X_k}(x_k)$ can be expressed as

$$\begin{aligned} f_{X_k}(x_k) \approx \frac{1}{z} \Bigg(& \xi \prod_{i=1}^{k-1} \lambda_i(\mathcal{X}_i) \prod_{i=k+1}^{d} \lambda_i(\mathcal{X}_i) \\ & + \left(\int_{\mathcal{X}_1} \lambda_1(x_1) \boldsymbol{G}_1^2(x_1) \, \mathrm{d}x_1 \right) \cdots \left(\int_{\mathcal{X}_{k-1}} \lambda_{k-1}(x_{k-1}) \boldsymbol{G}_{k-1}^2(x_{k-1}) \, \mathrm{d}x_{k-1} \right) \\ & \lambda_k(x_k) \boldsymbol{G}_k^2(x_k) \\ & \left(\int_{\mathcal{X}_{k+1}} \lambda_{k+1}(x_{k+1}) \boldsymbol{G}_{k+1}^2(x_{k+1}) \, \mathrm{d}x_{k+1} \right) \cdots \left(\int_{\mathcal{X}_d} \lambda_d(x_d) \boldsymbol{G}_d^2(x_d) \, \mathrm{d}x_d \right) \Bigg). \end{aligned} \tag{2.27}$$

2.2.1 Marginal Probability Distributions

We compute the marginal probability distributions by a procedure to which Cui et al. [9] refer to as backwards marginalisation, see Prop. 2, and to which we add the forward marginalisation, see Prop. 1. By now you have refered to multiple marginal distributions, so you'll need to be more clear. This is similar to the left and right orthogonalisation of TT why abbreviate? cores [38, 37]. The backwards marginalisation provides us with the

coefficient matrices $\boldsymbol{B}_k$, while the forward marginalisation gives the coefficient matrices $\boldsymbol{R}_{\text{pre},k}$. These matrices enable the efficient evaluation of marginal functions since they integrate over the coordinates either left or right of the k-th dimension, as in [9]. are they marginal functions or marginal distributions. Pick a language and stick to it. In doing so, we define the mass matrix $\boldsymbol{M}_k \in \mathbb{R}^{n_k \times n_k}$ as

$$\boldsymbol{M}_k[i,j] = \int_{\mathcal{X}_k} \phi_k^{(i)}(x_k)\phi_k^{(j)}(x_k)\boldsymbol{\lambda}(x_k)\,\mathrm{d}x_k, \quad i,j = 1,\dots,n_k, \text{ [9, Eq. 22]}, \tag{2.28}$$

where $\{\phi_k^{(i)}(x_k)\}_{i=1}^{n_k}$ denotes the set of basis functions for the k-th coordinate. The proposition used to compute $\boldsymbol{B}_k$, stated in Prop. 1, is adapted directly from [9].

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

$$f_{X_k}(x_k) \approx \frac{1}{z}\left(\xi \prod_{i=1}^{k-1}\boldsymbol{\lambda}_i(X_i)\prod_{i=k+1}^{d}\boldsymbol{\lambda}_i(X_i) + \sum_{l_{k-1}=1}^{r_{k-1}}\sum_{l_k=1}^{r_k}\left(\sum_{i=1}^{n}\phi_k^{(i)}(x_k)\boldsymbol{D}_k[l_{k-1},i,l_k]\right)^2\right)\boldsymbol{\lambda}_k(x_k), \tag{2.29}$$

where $\boldsymbol{D}_k \in \mathbb{R}^{r_{k-1}\times n\times r_k}$ and given as

$$\boldsymbol{D}_k[l_{k-1},i,l_k] = \sum_{\alpha_{k-1}=1}^{r_{k-1}} \boldsymbol{R}_{\text{pre},k-1}[l_{k-1},\alpha_{k-1}]\boldsymbol{B}_k[\alpha_{k-1},i,l_k]\,, \tag{2.30}$$

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

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

$$f_{X_1}(x_1) \approx \frac{1}{z}\left(\xi\prod_{i=2}^{d}\boldsymbol{\lambda}_i(\mathcal{X}_i) + \sum_{l_1=1}^{r_1}\left(\sum_{i=1}^{n}\phi_1^{(i)}(x_1)\boldsymbol{D}_1[i,l_1]\right)^2\right)\boldsymbol{\lambda}_1(x_1) \text{ [9, Eq. 30]}, \tag{2.31}$$

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

$$f_{X_d}(x_d) \approx \frac{1}{z}\left(\xi\prod_{i=1}^{d-1}\boldsymbol{\lambda}_i(\mathcal{X}_i) + \sum_{l_{d-1}=1}^{r_{d-1}}\left(\sum_{i=1}^{n}\phi_1^{(i)}(x_1)\boldsymbol{D}_d[l_{d-1},i]\right)^2\right)\boldsymbol{\lambda}_d(x_d), \tag{2.32}$$

where $\boldsymbol{D}_d[l_{d-1},i] = \boldsymbol{B}_{\text{pre},d}[l_{d-1},i,\alpha_{d+1}]$ and $\alpha_{d+1} = 1$. Note that in practice we calculate z numerically within the process of computing the marginals so that $\sum f_{X_k}(x_k) = 1$ and for Cartesian basis $\boldsymbol{M}_k = \text{diag}(\boldsymbol{\lambda}_k(\mathcal{X}_k))$ where we set $\boldsymbol{\lambda}(x) = 1$.

Proposition 1 (Backwards Marginalisation as in [9])**:** 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 $\boldsymbol{B}_k \in \mathbb{R}^{r_{k-1} \times n_k \times r_k}$, which we need for defining the marginal function $f_{X_k}(x_k)$ or to draw samples from $\tilde{\pi}(\boldsymbol{x})$ via the squared IRT scheme (see Alg. Box 1):

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

$$\boldsymbol{C}_k[\alpha_{k-1}, \tau, l_k] = \sum_{i=1}^{n_k} \boldsymbol{B}_k[\alpha_{k-1}, i, l_k] \boldsymbol{L}_k[i, \tau] \text{[9, Eq. (27)]}. \tag{2.33}$$

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

$$\boldsymbol{Q}_k \boldsymbol{R}_k = (\boldsymbol{C}_k^{(R)})^\top \text{ [9, Eq. 28]}. \tag{2.34}$$

3. Compute the new coefficient tensor:

$$\boldsymbol{B}_{k-1}[\alpha_{k-2}, i, l_{k-1}] = \sum_{\alpha_{k-1}=1}^{r_{k-1}} \boldsymbol{A}_{k-1}[\alpha_{k-2}, i, \alpha_{k-1}] \boldsymbol{R}_k[l_{k-1}, \alpha_{k-1}] \text{ [9, Eq. 29]}. \tag{2.35}$$

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

$$\boldsymbol{C}_{\text{pre},k}[\alpha_{k-1}, \tau, l_k] = \sum_{i=1}^{n_k} \boldsymbol{L}_k[i, \tau] \boldsymbol{B}_{\text{pre},k}[\alpha_{k-1}, i, l_k]. \tag{2.36}$$

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

$$\boldsymbol{Q}_{pre,k} \boldsymbol{R}_{\text{pre},k} = (\boldsymbol{C}_{\text{pre},k}^{(R)}). \tag{2.37}$$

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

$$\boldsymbol{B}_{\text{pre},k+1}[l_{k+1}, i, \alpha_{k+1}] = \sum_{\alpha_k=1}^{r_k} \boldsymbol{R}_{\text{pre},k}[l_{k+1}, \alpha_k] \boldsymbol{A}_{k+1}[\alpha_k, i, \alpha_{k+1}]. \tag{2.38}$$

2.2.2 Sampling from a TT Approximation

Instead of evaluating marginal functions for quadrature, we can draw samples from the approximated function in the TT format via the inverse Rosenblatt transform, as in [13], to preserve the correlation structure of the parameters. what does this mean? If you draw from a distribution, it automatically has the right correlation structure. or are you trying to say something else, the I can't work out. Since we approximate the square root of the target function, Cui et. al. [9] call that the squared inverse Rosenblatt transform (SIRT). the 'we' seems quite inappropriate here, as it's Cui et al that call it SIRT.

Algorithm 1: Squared Inverse Rosenblatt Transform (SIRT)

Input: seeds $\{\boldsymbol{u}^{(1)}, \dots, \boldsymbol{u}^{(N)}\} \sim \mathcal{U}(0,1)^d$ and $\boldsymbol{B}_1, \dots, \boldsymbol{B}_d$ from Prop. 1
for $s = 1, \dots, N$ **do**
  for $k = 1, \dots, d$ **do**
    compute normalised PDF $f_{X_k|X_{<k}}(x_k|x_{k-1}^{(s)}, \dots, x_1^{(s)})$, Eq. 2.40
    compute cumulative distribution function $F_{X_k|X_{<k}}(x_k)$, Eq. 2.39,
    project sample $x_k^{(s)} = F^{-1}_{X_k|X_{<k}}(u_k^{(s)})$
    interpolate $\boldsymbol{G}_k(x_k^{(s)})$, Eq. 2.41
    update $\boldsymbol{G}_{\leq k}(x_{\leq k}^{(s)}) = \boldsymbol{G}_{<k}(x_{<k}^{(s)})\boldsymbol{G}_k(x_k^{(s)})$
  end for
end for
Output: samples $\{\boldsymbol{x}^{(1)}, \dots, \boldsymbol{x}^{(N)}\}$, where each $\boldsymbol{x}^{(s)} \in \mathbb{R}^d$ for $s = 1, \dots, N$

Given the Backward marginals $\boldsymbol{B}_1, \dots, \boldsymbol{B}_d$ as in Prop. 1, we draw N uniformly distributed seeds $\{\boldsymbol{u}^{(1)}, \dots, \boldsymbol{u}^{(N)}\} \sim \mathcal{U}(0,1)^d$, where each $\boldsymbol{u}^{(s)}$ is d-dimensional for $s = 1, \dots, N$. Then we calculate the first marginal $f_{X_1}(x_1)$ as in Eq. 2.31 and normalise with $z = \int_{\mathcal{X}_1} f_{X_1}(x_1)dx_1$. OK, another task – remove all but one 'we' per page, and even then you will have 100, which is too many. FYI, I counted 397 occurrences of 'we' in this text – so you only have to rewrite 297 sentences! Holy Toledo, here's one to zap. and another. Is this a novel about Lennart's adventures in MCMC for 10 year olds? Next, we compute the cumulative distribution function (CDF) $F_{X_1}(x_k) = \int_{-\infty}^{x_k} f_{X_1}(\hat{x}_1)d\hat{x}_1$, which for the general case is given as:

$$F_{X_k|X_{<k}}(x_k) = \int_{-\infty}^{x_k} f_{X_k|X_{<k}}(\hat{x}_k|x_{k-1}, \dots, x_1)\,\mathrm{d}\hat{x}_k \text{ [9, Eq. 17]}; \tag{2.39}$$

and project the seed $u_k^{(s)}$ on the parameter space, so that we generate a sample $x_k^{(s)} =$

$F^{-1}_{X_k|X_{<k}}(u_k^{(s)})$. Argghhhh! The "conditional marginal" is given as:

$$f_{X_k|X_{<k}}(x_k|x_{k-1}^{(s)},\dots,x_1^{(s)}) \approx \frac{1}{z}\Bigg(\xi \prod_{i=k+1}^{d} \boldsymbol{\lambda}_i(X_i) + \sum_{l_k=1}^{r_k}\Bigg(\sum_{i=1}^{n}\phi_k^{(i)}(x_k^{(s)})\Bigg(\sum_{\alpha_{k-1}=1}^{r_{k-1}} \boldsymbol{G}_{<k}^{(\alpha_{k-1})}(x_{<k}^{(s)})\boldsymbol{B}_k[\alpha_{k-1},i,l_k]\Bigg)\Bigg)^2\Bigg)\boldsymbol{\lambda}_k(x_k) \text{ [9, Eq. 31]}, \tag{2.40}$$

where we marginalise over the dimensions $k+1,\dots,d$ via $\boldsymbol{B}_k$ and condition on the previous $k-1$ samples through the product $\boldsymbol{G}_k(x_k^{(s)}) \in \mathbb{R}^{1\times r_{k-1}}$ to preserve the correlation structure. God dammit, another pesky we. Between grid points i and $i+1$, we approximate using a piecewise polynomial interpolation

$$\boldsymbol{G}_k(x_k^{(s)}) \approx \frac{x_k^{(s)}-x_k^{(i)}}{x_k^{(i+1)}-x_k^{(i)}}\boldsymbol{G}_k(x_k^{(i+1)}) + \frac{x_k^{(i+1)}-x_k^{(s)}}{x_k^{(i+1)}-x_k^{(i)}}\boldsymbol{G}_k(x_k^{(i)}), \tag{2.41}$$

for $x_k^{(i)} \le x_k^{(s)} \le x_k^{(i+1)}$ as in [13] for the next "conditional marginal".

We repeat the procedure for each $u_k^{(s)} \in \boldsymbol{u}^{(s)}$ to produce samples $\boldsymbol{x}^{(s)} \sim f_X(\boldsymbol{x})$, as summarised in Alg. Box 1. The procedure is repeated ... (the 'we' is entirely redundant, and distracting.)

Metropolis-Hastings – correction step

Since the samples by the SIRT scheme are generated from an approximation, it is sensible to correct those using a Metropolis-Hastings (MH) importance step. In doing so, we

Algorithm 2: MH correction step

Input: samples $\{\boldsymbol{x}^{(1)},\dots,\boldsymbol{x}^{(N+1)}\}$, where each $\boldsymbol{x}^{(s)} \in \mathbb{R}^d$ for $s = 1,\dots,N+1$
for $s = 1,\dots,N$ **do**
  compute MH ratio $\frac{w^{(s+1)}}{w^{(s)}} = \frac{\pi(\boldsymbol{x}^{(s+1)})}{\pi(\boldsymbol{x}^{(s)})}\frac{f_X(\boldsymbol{x}^{(s)})}{f_X(\boldsymbol{x}^{(s+1)})}$
  compute acceptance probability $\alpha = \min(w^{(s+1)}/w^{(s)}, 1)$
  Draw $u \sim \mathcal{U}(0,1)$
  if $\alpha \ge u$ **then**
    Accept and set $\boldsymbol{x}_{\text{MH}}^{(s+1)} = \boldsymbol{x}^{(s+1)}$
  else
    Reject and keep $\boldsymbol{x}_{\text{MH}}^{(s+1)} = \boldsymbol{x}^{(s)}$
  end if
end for
Output: corrected sample chain $\{\boldsymbol{x}_{\text{MH}}^{(1)},\dots,\boldsymbol{x}_{\text{MH}}^{(N)}\}$, where each $\boldsymbol{x}_{\text{MH}}^{(s)} \in \mathbb{R}^d$ for $s = 1,\dots,N$

compute the acceptance probability $\alpha = \min(w^{(s+1)}/w^{(s)}, 1)$, where

$$w(x) = \frac{\pi(\boldsymbol{x})}{f_X(\boldsymbol{x})} = \frac{\pi(\boldsymbol{x})}{\xi + \tilde{g}(\boldsymbol{x})^2} \tag{2.42}$$

is the importance ratio. Note that since we calculate the ratio $w^{(s+1)}/w^{(s)}$, the normalising constants cancel. In practice, we calculate the importance ratio in the log space, where $\log f_X(\boldsymbol{x}) = \log f_{X_1}(x_1) + \log f_{X_2|X_1}(x_2|x_1) + \cdots + \log f_{X_k|X_{<k}}(x_k|x_{k-1}, \ldots, x_1)$ is given as in Eq. 2.40, see [13]. We refer to this as the SIRT-MH scheme, which in theory provides the corrected chain $\{\boldsymbol{x}_{\mathrm{MH}}^{(1)}, \ldots, \boldsymbol{x}_{\mathrm{MH}}^{(N)}\} \sim \pi(\boldsymbol{x})$.

2.2.3 Error of the TT Approximation

A straightforward way to assess the average relative error from the TT approximation is to calculate the relative root mean squared (RMS) error

$$\left(\frac{\int_{\mathcal{X}} (\pi(\boldsymbol{x}) - (\xi + \tilde{g}(\boldsymbol{x})^2))^2 \boldsymbol{\lambda}(\boldsymbol{x}) \, \mathrm{d}\boldsymbol{x}}{\int_{\mathcal{X}} \pi(\boldsymbol{x})^2 \boldsymbol{\lambda}(x) \, \mathrm{d}\boldsymbol{x}} \right)^{1/2} = \frac{\|\pi(\boldsymbol{x}) - (\xi + \tilde{g}(\boldsymbol{x})^2)\|_{L^2_{\boldsymbol{\lambda}}(\mathcal{X})}}{\|\pi(\boldsymbol{x})\|_{L^2_{\boldsymbol{\lambda}}(\mathcal{X})}} \,. \tag{2.43}$$

The RMS is approximated by

$$\left(\frac{1}{N} \sum_{i=1}^{N} \left(\pi(\boldsymbol{x}^{(i)}) - (\xi + \tilde{g}(\boldsymbol{x}^{(i)})^2) \right)^2 \boldsymbol{\lambda}(\boldsymbol{x}^{(i)}) \right)^{1/2} \approx \left(\int_{\mathcal{X}} \left(\pi(\boldsymbol{x}) - (\xi + \tilde{g}(\boldsymbol{x})^2) \right)^2 \boldsymbol{\lambda}(\boldsymbol{x}) \, \mathrm{d}\boldsymbol{x} \right)^{1/2} \tag{2.44}$$

and similarly $\int_{\mathcal{X}} \pi(\boldsymbol{x})^2 \boldsymbol{\lambda}(\boldsymbol{x}) \, \mathrm{d}\boldsymbol{x}$.

Absolute error bound

The Wasserstein distance is the infimum over all couplings between two probability distributions with respect to some distance measure. If large errors occur in regions with low probability the RMS is sensible to those, whereas the Wasserstein distance includes probability values and weights the distance measures accordingly. The Kantorovich-Rubinstein duality, as in [58, 1], says that the 1-Wasserstein distance is equal to the supremum of differences in expectations over all 1-Lipschitz functions h between two probability distributions. So the 1-Wasserstein distance provides an upper absolute error bound.

The 1-Wasserstein distance is defined as

$$W_1(\pi, \tilde{\pi}) = \inf_{\nu \in \Pi(\pi, \tilde{\pi})} \int_{\mathcal{X} \times \mathcal{X}} c_{\mathcal{X}}(\boldsymbol{x}, \tilde{\boldsymbol{x}}) \, \nu(\boldsymbol{x}, \tilde{\boldsymbol{x}}) d\boldsymbol{x} d\tilde{\boldsymbol{x}} \,, \tag{2.45}$$

where ν couples $\boldsymbol{x}$ and $\tilde{\boldsymbol{x}}$ so that the integral over the distance $c_{\mathcal{X}}(\boldsymbol{x}, \tilde{\boldsymbol{x}})$ weighted by the probability measures π and $\tilde{\pi}$ is the greatest lower bound of all integrals with respect to ν in the set of all couplings $\Pi(\pi, \tilde{\pi})$. Often ν is called a transport plan, where $c_{\mathcal{X}}(\boldsymbol{x}, \tilde{\boldsymbol{x}})$ is the (ground) cost function, and $\nu(\boldsymbol{x}, \tilde{\boldsymbol{x}})$ is related to the mass which has to be

transported and the 1-Wasserstein distance is the earth mover distance. On the other hand (Kantorovich-Rubinstein duality) the 1-Wasserstein distance

$$W_1(\pi,\tilde{\pi}) = \sup_{h(\boldsymbol{x});\, c_{\mathcal{Y}}(h(\boldsymbol{x}),h(\tilde{\boldsymbol{x}}))\leq c_{\mathcal{X}}(\boldsymbol{x},\tilde{\boldsymbol{x}})} \left\{ \int_{\mathcal{X}} h(\boldsymbol{x}) d\pi(\boldsymbol{x}) - \int_{\mathcal{X}} h(\tilde{\boldsymbol{x}}) d\tilde{\pi}(\tilde{\boldsymbol{x}}) \right\} \tag{2.46}$$

$$= \sup_{h(\boldsymbol{x});\, c_{\mathcal{Y}}(h(\boldsymbol{x}),h(\tilde{\boldsymbol{x}}))\leq c_{\mathcal{X}}(\boldsymbol{x},\tilde{\boldsymbol{x}})} \left\{ \mathop{\mathbb{E}}_{\boldsymbol{x}\sim\pi}\big[h(\boldsymbol{x})\big] - \mathop{\mathbb{E}}_{\tilde{\boldsymbol{x}}\sim\tilde{\pi}}\big[h(\tilde{\boldsymbol{x}})\big] \right\}. \tag{2.47}$$

is the lowest upper bound of differences in expectations over all 1-Lipschitz function $h(\boldsymbol{x}) : \mathcal{X} \to \mathcal{Y}$ in between the two distributions π and $\tilde{\pi}$, with the distance measure $c_{\mathcal{X}}$ on the set $\mathcal{X}$ forming the metric space $(\mathcal{X}, c_{\mathcal{X}})$ and similarly the metric space $(\mathcal{Y}, c_{\mathcal{Y}})$.

For two sample sets $\{\boldsymbol{x}^{(1)},\dots,\boldsymbol{x}^{(N)}\} \sim \pi$ and $\{\tilde{\boldsymbol{x}}^{(1)},\dots,\tilde{\boldsymbol{x}}^{(M)}\} \sim \tilde{\pi}$ the calculation of the Wasserstein distance becomes an optimisation problem that is to find the best coupling of samples weighted by their distribution value according to an appropriate distance measure [16], which we set to $c_{\mathcal{X}}(\boldsymbol{x},\tilde{\boldsymbol{x}}) = \|\boldsymbol{x}-\tilde{\boldsymbol{x}}\|_{L^2}$. More specifically,

$$W_1(\pi,\tilde{\pi}) = \min_{\nu\in\Pi(\pi,\tilde{\pi})} \sum_{j=1}^{M}\sum_{i=1}^{N} \nu_{ij} \|\boldsymbol{x}^{(i)} - \tilde{\boldsymbol{x}}^{(j)}\|_{L^2}\,, \tag{2.48}$$

where the transport plan $\nu \in \mathbb{R}_{\geq 0}^{N\times M}$ defines the coupling $\nu_{ij} \in \nu$ as $\nu_{ij} \coloneqq \pi(\boldsymbol{x}^{(i)})\tilde{\pi}(\tilde{\boldsymbol{x}}^{(j)})$ similar to [16, Eq. 3.166]. Additionally we require that $\sum_{i=1}^{N}\pi(\boldsymbol{x}^{(i)}) = \sum_{j=1}^{M}\tilde{\pi}(\tilde{\boldsymbol{x}}^{(j)}) = 1$. This gives us an upper bound of the absolute error between the expected value of any 1-Lipschitz function h.

3
The Forward Model

In this chapter, we present the forward model to which we apply all our methodology. We follow the Michelson interferometer for passive atmospheric sounding (MIPAS) handbook [42] and simulate data according to an idealised cloud-free atmosphere in local thermodynamic equilibrium, assuming a measurement instrument with infinite spectral resolution and no pointing errors. This is a simplified forward model and we do not include any other instrument-specific details, such as sensor area or antenna response, as they are not available to us.

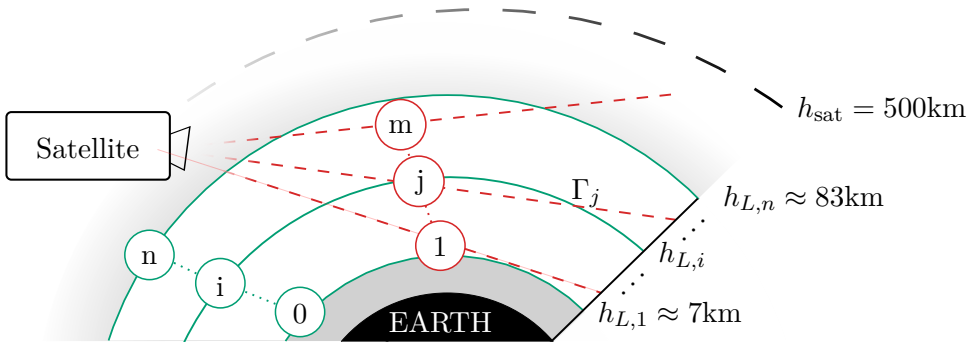


Figure 3.1: Schematic of measurement and analysis geometry, not to scale. The stationary satellite, at a constant height h_{sat} above Earth, takes m measurements along its line-of-sight defined by the line Γ_j. Each measurement has a limb height ℓ_j, $j = 1, 2, \ldots, m$ defined as the closest distance of Γ_j to the Earth's surface. Between $h_{L,0} \approx 7\text{km}$ and $h_{L,n} \approx 83\text{km}$, the atmosphere is discretised into n layers as illustrated by the solid green lines.

A satellite at a constant height h_{sat} points through the atmosphere (limb-sounding) and measures thermal radiation of gas molecules along its straight line of sight Γ_j, see Figure 3.1. One measurement of the thermal radiation of one specific molecule, in our case ozone, denoted by the ozone volume mixing ratio (VMR) $x(r)$ at distance r from

the satellite, at the wave number ν, is given by the path integral

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

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

which is the radiative transfer equation (RTE) [42]. For more information on the processes within the atmosphere for ozone, we refer to [30]. We define a tangent height h_{ℓ_j} and Γ_j for each $j = 1, 2, \ldots, m$, so that the data vector $\boldsymbol{y} \in \mathbb{R}^m$ including some additive noise η_j. Within the atmosphere, the number density $p(T)/(k_{\mathrm{B}} T(r))$ 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

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

is dependent on the line intensity $L(\nu, T_{\mathrm{ref}})$ at reference temperature $T_{\mathrm{ref}} = 296K$, the lower-state energy E'' in cm^{-1} of the targeted transition and the second radiation constant $c_2 := hc/k_{\mathrm{B}} \approx 1.44\mathrm{cmK}$ as in the HITRAN database [23], with Planck's constant h and speed of light c. Since we assume that the measurement device has a negligible frequency window, we neglect line broadening around ν for the calculations of $L(\nu, T_{\mathrm{ref}})$, which would normally be modelled as a convolution of the normalised Lorentz profile (collisional/pressure broadening) and the normalised Doppler (thermal broadening) profile [42]. Additionally, we target one specific molecule and calculate $k(\nu, T)$ accordingly, which usually would involve summing the individual absorption constants for multiple radiating molecules weighted by their respective VMR [42]. The total internal partition function is given as:

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

with the statistical weight g'' for the lower and g' for the upper energy state (also called the degeneracy factors) accounting for the molecule's non-rotational and rotational energy states (see [53]), and the upper state energy $E' = E'' + \nu$. Under the assumption of local thermodynamic equilibrium (LTE), the black body radiation acts as a source function

$$B(\nu, T) = \frac{2hc^2\nu^3}{\exp\{\frac{c_2\nu}{T}\} - 1}. \tag{3.5}$$

For fundamentals on the RTE, we recommend [49, Chapter 1], and for a more comprehensive model, we refer to [41].

To calculate the integrals in Eq. 3.1 and Eq. 3.2 numerically, 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 $h_{L,0}$ and $h_{L,n}$. Then the ozone VMR $\boldsymbol{x} = \{x_1, x_2, \ldots, x_n\} \in \mathbb{R}^n$, pressure $\boldsymbol{p} = \{p_1, p_2, \ldots, p_n\} \in \mathbb{R}^n$ and temperature $\boldsymbol{T} = \{T_1, T_2, \ldots, T_n\} \in \mathbb{R}^n$, as well as all other height dependent parameters, are discretised profiles with constant values in between the heights $h_{L,i-1}$ and $h_{L,i}$. Above $h_{L,n}$ and below $h_{L,0}$, the ozone VMR is set to zero, so no signal can be obtained. We evaluate the integral in Eq. (3.1) for one noise-free measurement $\boldsymbol{A_j}(\boldsymbol{p}, \boldsymbol{T}, \boldsymbol{x})$, using the trapezoidal rule. Here, each entry of $\boldsymbol{A}(\boldsymbol{p}, \boldsymbol{T}, \boldsymbol{x}) \in \mathbb{R}^m$ includes multiple evaluations of the integral in Eq. 3.2 to calculate the absorption $\tau(r)$. The data vector

$$\boldsymbol{y} = \boldsymbol{A}(\boldsymbol{x}) + \boldsymbol{\eta} \tag{3.6}$$

includes an additive noise vector $\boldsymbol{\eta} \in \mathbb{R}^m$, where we define the non-linear forward model as $\boldsymbol{A}(\boldsymbol{x}) \coloneqq \boldsymbol{A}(\boldsymbol{x}, \boldsymbol{p}, \boldsymbol{T}) \in \mathbb{R}^m$ for brevity. Similarly, we define $\boldsymbol{A}_L$, which denotes the linear forward model matrix and neglects absorption (e.g. set $\tau = 1$ in Eq. (3.2)) and enables matrix-vector multiplication $\boldsymbol{A}_L\boldsymbol{x}$ to compute noise-free linear data.

Further, we classify the inverse problem as a weakly non-linear inverse problem, because neglecting the absorption changes the measurements only slightly (about 1%, see Sec. **??**).

3.1 Understanding the Forward Model

understanding the forward map Before simulating some data, we provide a quick and intuitive way of assessing if the data collection is effective, how much information is passed through the forward model, and how the signal-to-noise ratio (SNR) and the measurement strategy affects that information. One way of doing this is via a singular value decomposition (SVD) of the forward model matrix

$$\boldsymbol{A}_L = \sum_{i=1}^{r} \boldsymbol{u}_i \sigma_i \boldsymbol{v}_i^T = \boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^T \tag{3.7}$$

where $r = \min\{m, n\}$ for a forward model matrix $\boldsymbol{A}_L \in \mathbb{R}^{m \times n}$. Consider noise-free measurements $\boldsymbol{A}_L\boldsymbol{x}$ for a satellite at a fixed height of $h_{\text{sat}} = 500\text{km}$ above sea level. Our main objective is to measure ozone $\boldsymbol{x}$, so our forward model $\boldsymbol{A}_L$ includes temperature and pressure, the latter is dominant, see Fig. 6.4, decreases exponentially in height and hence does affect the information passed through the model. If the pressure is high, the signal is large. If the pressure is low, the signal is low and the data is noise-dominated.

The SVD gives us information on how information is picked up from the parameter space by the forward model, described through the right singular values $\boldsymbol{v}_i$. The singular values σ_i, ordered in size from the largest σ_1 to the smallest σ_r, weigh that information from the right singular values to the left singular values $\boldsymbol{u}_i$, which project onto the data space. For a large singular value, we can say that the forward model is informative

about parameter structures according to the corresponding right singular vector and vice versa. WQhat is the vice versa? Not clear what you are saying. Further, for very small singular values $\sigma_i \ll \sigma_1/\text{SNR}$ below the RMS noise level or the noise standard deviation (STD), we can introduce an effective rank $r_{\text{eff}} \leq r$. Then, the parameter space spanned by $\{\boldsymbol{v}_{r_{\text{eff}}+1}, \ldots, \boldsymbol{v}_r\}$ is noise-dominated in the corresponding data space, see Figure 3.6. This is based on the rough assumption that if we define the SNR as don't put citations in equations – it's very weird.

$$\text{SNR} := \frac{\max(y)}{\text{STD noise}} = \frac{\text{peak signal}}{\text{RMS noise}} [18], \tag{3.8}$$

then the maximum singular value $\max(y) \approx \sigma_1$ and the information transmitted through the forward model corresponds roughly to the singular values $\sigma_i \gtrsim \max(y)/\text{SNR}$. See [57] for a more comprehensive analysis.

why is this plot here before you defined the cases? That's not remotely OK. oh

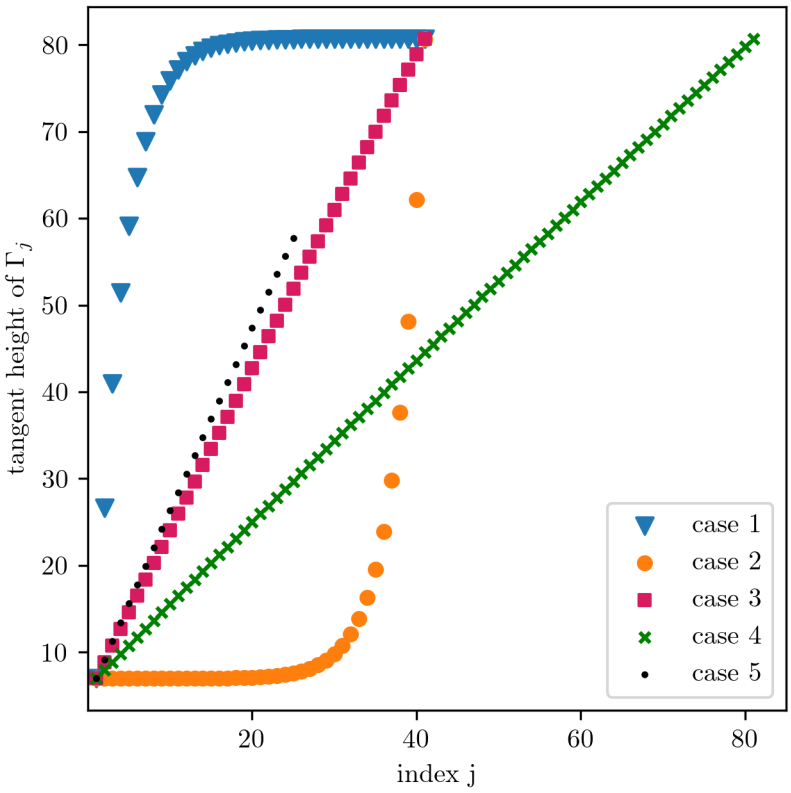


Figure 3.2: Tangent heights for five different sequences of measurements.

no, I had hpped this section would be 'we' free. where do you explain why this SNR has a line here?, Were is the clear definition of these cases? I could not find it. Next, we plot the singular values for five different measurement scenarios, where we either measure at equidistance-spaced tangent heights or collect more data from high signal

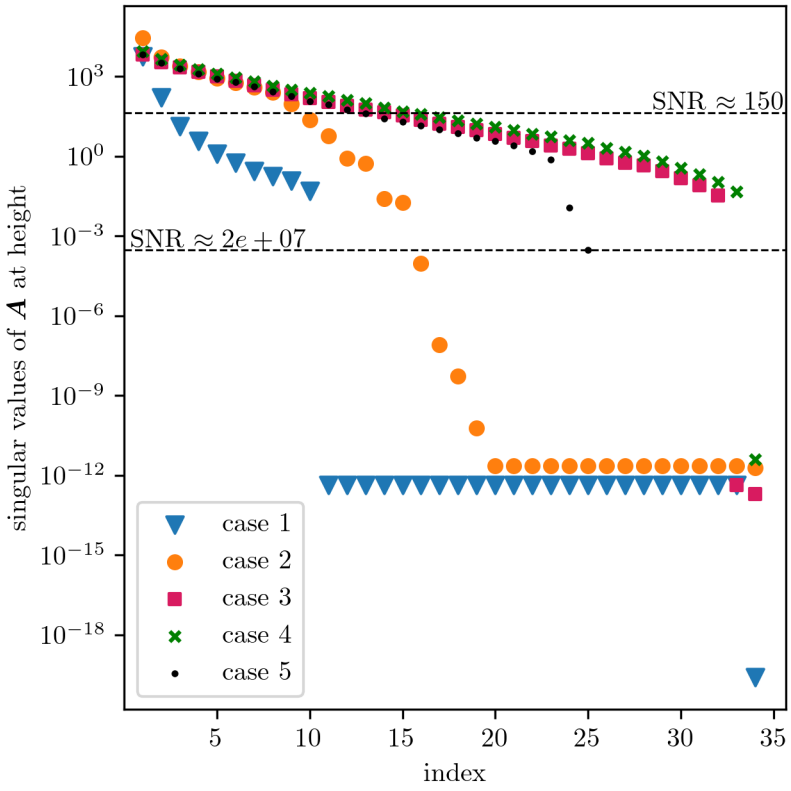


Figure 3.3: Singular values of the forward model matrix for different sequences of measurements. The corresponding tangent heights of the test cases are plotted in Fig. 3.2. The dotted vertical line marks where the singular values are dominated by noise according to a specific SNR.

regions at low altitudes or from low signal regions at high altitudes. We assess the number of singular values above and below a certain SNR visually to determine which of the tested cases is most effective. Another, they are multiplying like rabbits. Or flies. Which is more annoying? We start with case 3 in Fig. 3.2 where measurements are spaced according to a pointing accuracy of 150arc sec, given to us by the team of the University of New South Wales Canberra Space [15]. more sentence creep. The pointing accuracy determines how well the satellite can point in a certain direction and, hence, roughly the spacing in between two measurements. Actually, you should calculate this from the MIPAS specifications, or similar. It claims 1-3km resolution, presumably at a distance of 500km. So, what's that in arc seconds? It's also related to your vertical discretization, or, more to the point, your vertical discretization is determined by this. The corresponding singular values are plotted in Fig. 3.3, of which the first 25 decrease linearly in log-space and about 10-15 singular values lie above the SNR. where is this spacing? At the satellite? A spacing in time? On the planet Mars. Be specific. All the way through, you need to tighten up loose statements like this. In comparison, if we measure a lot of times in regions where the data is noise-dominated (high altitude),

case 1, we do obtain more information since the singular values decrease rapidly. Oh come on, you are showing pictures of singular values without specifying exactly what the forward map and discretization is. That's not acceptable. If these are intended to be indicative, you need to be a damn site clearer about what they represent, and are trying to show. Measuring lots of times at low altitudes, where the data is informative, and less at higher altitudes, case 2, does not seem optimal either, as we observe one larger singular value, but the other singular values decrease faster compared to case 3. Now consider case 4, where we double the number of measurements compared to case 3, we do get slightly larger singular values, but not so significantly that it would be worth the engineering effort required to achieve that. The measurements with equidistance-spaced tangent height seem to be most effective. why? Explain your reasoning. If you don't say something specific, quantitative, it's just waffle. By exploratory analysis, we find that we can tolerate a slightly larger distance between tangent heights (pointing accuracy of 175arc sec) than required by [15], see case 5. In that case, we also stop measuring when the signal is too noisy and decrease the number of measurements taken without losing crucial information. We note that if one wanted to obtain all information provided by the forward model, we would need a signal-to-noise ratio of roughly 10^7.

In principle, we show that it does not depend on how one measures; one cannot get more information by measuring more in regions where the information content is low or high. on the contrary - -you have just been arguing that it does matter. no, this makes sense if the goal is to reduce noise. This contradicts the current measurement setup on the AURA MLS [32], which reports high noise in lower atmospheric regions, due to thermal radiation from the earth, and measures more in those regions.

Consequently, we proceed with case 5 and plot the right singular vectors of the forward model versus height in the atmosphere to see where our model is most informative, or which structures of the parameter space are picked up by the model. The first 10 right singular vectors, in Fig. 3.4, corresponding to the 10 largest singular values, pick up parameter structures in lower atmospheric regions. So we can assume that, given some data, we will be able to provide good reconstructions of the parameter in lower altitudes. too colloquial. One picks up rubbish, not sure what it means in this context. Next, we plot the right singular vectors corresponding to the singular values σ_j for $j = 11, \ldots, 20$ in Fig. 3.5, where the noise starts to dominate the data. These singular values lie in regions around the SNR, see Fig. 3.3, and pick up values in the middle of the atmosphere. ditto Consequently, we expect a higher uncertainty of reconstructed parameter values in the middle atmospheric regions. The singular vectors corresponding to the last 10 singular values pick up parameter structures in higher altitudes, but since the singular values are very small, we will not be able to retrieve those structures. More specifically, the retrieved parameter values at higher altitudes will be mostly determined by the prior or, in the case of regularisation, by the regulariser [57].

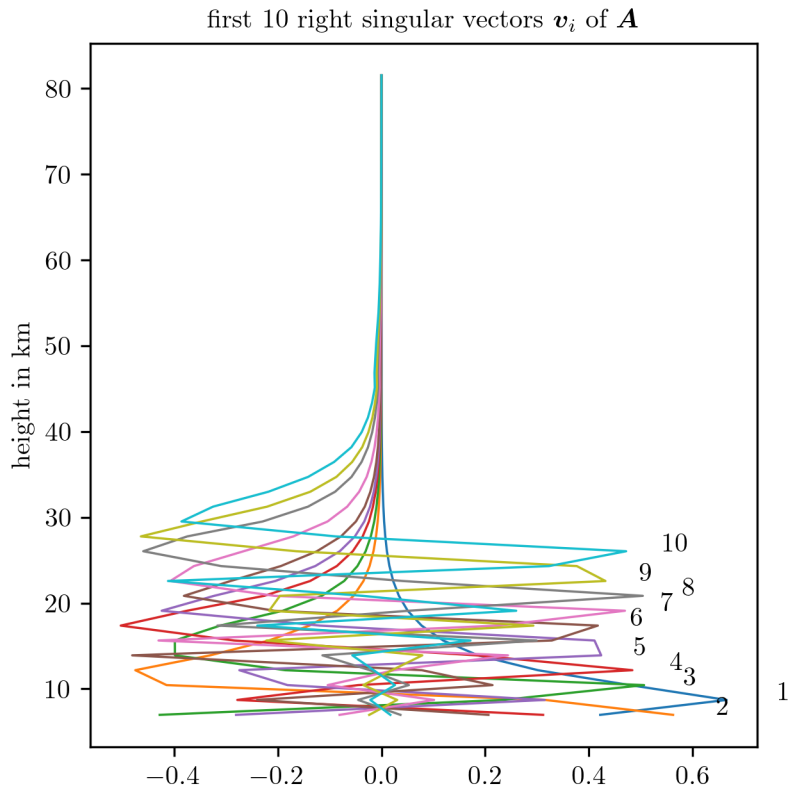


Figure 3.4: First 10 right singular vectors of the forward model matrix for measurements case 5 in Fig. 3.2. These singular vectors correspond to high singular values of the forward model in Fig. 3.3.

3.2 Simulate Data based on a Ground Truth

As the ground truth for our methodology, we consider an ozone profile at distinct pressure values generated from some data [34] of the MLS on the Aura satellite within the Antarctic region. This ozone profile has a peak in the middle atmosphere and a second peak at higher altitudes, see Fig. 4.5, which seems to be a typical nighttime profile [30].

We recursively relate pressure p to geometric height h with the hydrostatic equilibrium equation

$$\frac{\mathrm{d}p}{p} = \frac{-gM}{R^*T}\mathrm{d}h\,, \tag{3.9}$$

starting with a pressure of 1013.25hPa at sea level. The acceleration due to gravity

$$g = g_0\left(\frac{r_0}{r_0+h}\right), \tag{3.10}$$

where the polar radius of 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

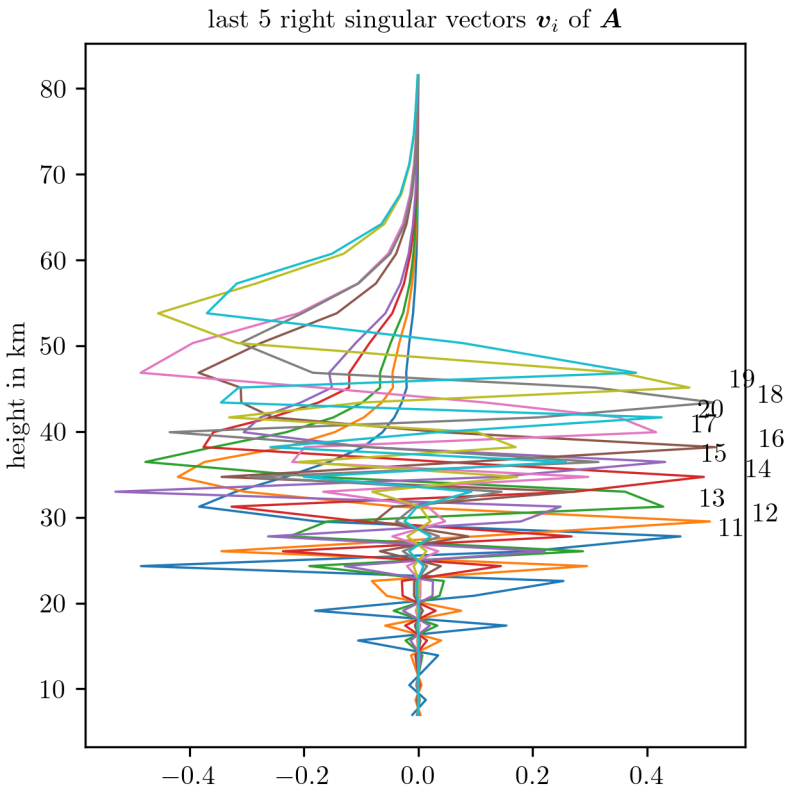


Figure 3.5: Right singular vectors with index $i = 11, \ldots, 19$ of the forward model matrix for measurements case 5 in Fig. 3.2. These singular vectors correspond to singular values in Fig. 3.3 where the noise level is similar to the data.

air is $M = 28.97\text{kg/kmol}$ [59]. This holds up to a geometric height of 86km, where we ignore a 0.04% non-linear change in M from 80km to 86km.

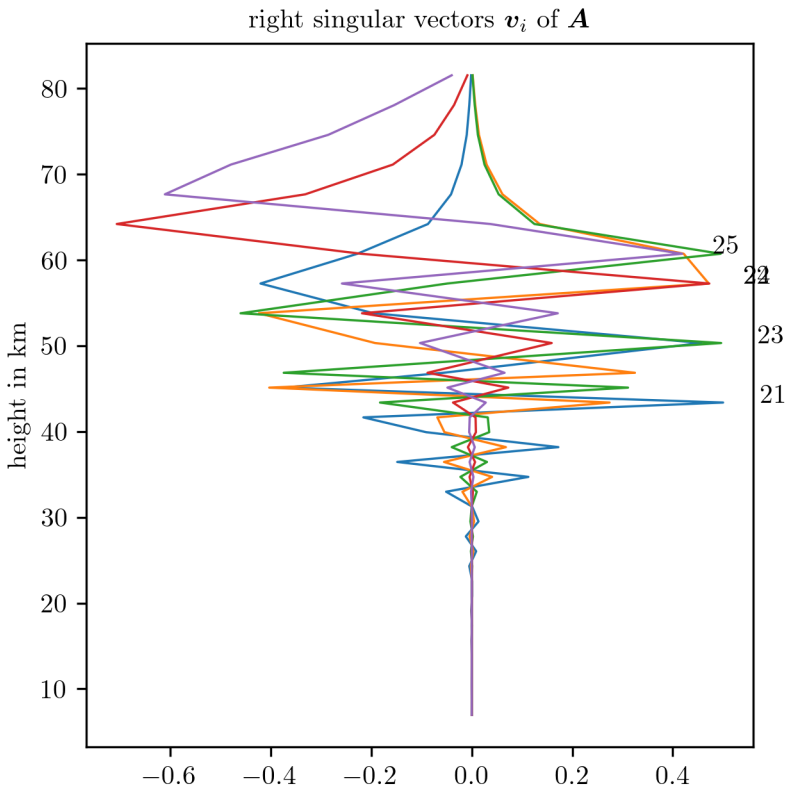


Figure 3.6: Last 10 right singular vectors of the forward model matrix for measurements case 5 in Fig. 3.2. These singular vectors correspond to small singular values of the forward model in Fig. 3.3 where the data is noise-dominated.

Following [59] we form the temperature function

$$
T(h) = \begin{cases}
T_0 & , h = 0 \\
T_0 + a_0 h & , 0 \le h < h_{T,1} \\
T_0 + a_0 h_{T,1} & , h_{T,1} \le h < h_{T,2} \\
T_0 + a_0 h_{T,1} + a_1(h_{T,2} - h_{T,1}) + a_2(h - h_{T,2}) & , h_{T,2} \le h < h_{T,3} \\
T_0 + a_0 h_{T,1} + a_1(h_{T,2} - h_{T,1}) & \\
\quad + a_2(h_{T,3} - h_{T,2}) + a_3(h - h_{T,3}) & , h_{T,3} \le h < h_{T,4} \\
T_0 + a_0 h_{T,1} + a_1(h_{T,2} - h_{T,1}) & \\
\quad + a_2(h_{T,3} - h_{T,2}) + a_3(h_{T,4} - h_{T,3}) + a_4(h - h_{T,4}) & , h_{T,4} \le h < h_{T,5} \\
T_0 + a_0 h_{T,1} + a_1(h_{T,2} - h_{T,1}) & \\
\quad + a_2(h_{T,3} - h_{T,2}) + a_3(h_{T,4} - h_{T,3}) + a_4(h_{T,5} - h_{T,4}) & \\
\quad + a_5(h - h_{T,5}) & , h_{T,5} \le h < h_{T,6} \\
T_0 + a_0 h_{T,1} + a_1(h_{T,2} - h_{T,1}) & \\
\quad + a_2(h_{T,3} - h_{T,2}) + a_3(h_{T,4} - h_{T,3}) + a_4(h_{T,5} - h_{T,4}) & \\
\quad + a_5(h_{T,6} - h_{T,5}) + a_6(h - h_{T,6}) & , h_{T,6} \le h \lesssim 86
\end{cases} \tag{3.11}
$$

subscript i	geometric height $h_{T,i}$ 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

Table 3.1: Definition of height depending temperature gradients.

with gradient and height values provided by [59] (see Tab. 3.1) which acts as the ground truth temperature profile (see Fig. 6.2).

Then we can compute a data vector $\boldsymbol{y} = \boldsymbol{A}(\boldsymbol{x}) + \boldsymbol{\eta}$, with $m = 30$ measurements determined by the satellite pointing accuracy of 175arc sec (see case 5 in Fig. 3.2), according to the RTE as in Eq. 3.1 and Eq. 3.2 using the trapezoidal integration rule. We assume an atmosphere between $h_{L,1} = 7$km and $h_{L,n} = 83.3$km with $n = 45$ equidistant layers (see Fig. 3.1). The height value $h_{L,i}$ for each layer $i = 1, \ldots, n$ is defined by the pressure values from [34] and the hydrostatic equilibrium equation, see Eq. 3.9. We target ozone at a frequency of 235.71GHz, which lies within the region where the MLS observes ozone [33, 63]. The corresponding wave number is $\nu = 7.86\text{cm}^{-1}$. We calculate the absorption constant $k(\nu, T)$ as in Eq. 3.2, following the high resolution transmission (HITRAN) database [23], which provides the line intensity $L(\nu, T_{\text{ref}})$ for the isotopologue $^{16}O_3$ with the AFGL Code 666. Lastly, we add independent and identically-distributed Gaussian noise $\boldsymbol{\eta} \sim \mathcal{N}(0, \gamma^{-1}\boldsymbol{I})$ so that the SNR = 150 (see Eq. 3.8) similar to [21], where a signal with a maximal spectral intensity of around 100K and a noise range of 0.4 to 1.6K is reported. We note that the methods used in this thesis will work with different SNRs or other frequencies.

We plot the data in Fig. 3.7, which is noise-dominated in higher altitudes and hence not sensitive to structures in the higher atmospheric regions. Now, given the data, we like to determine the posterior distributions over ozone $\boldsymbol{x}$, pressure $\boldsymbol{p}$ and temperature $\boldsymbol{T}$.

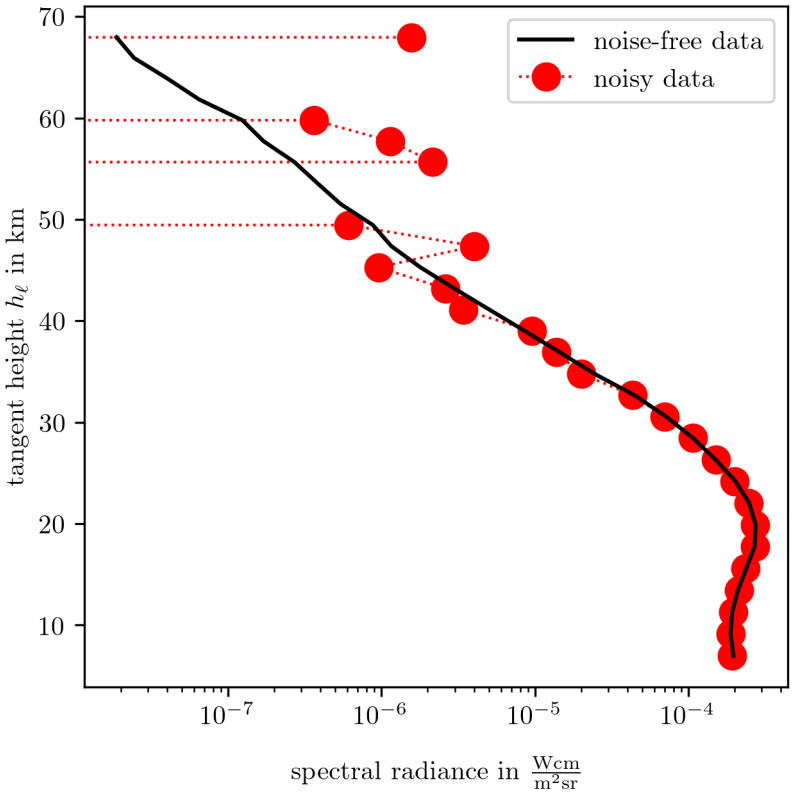


Figure 3.7: Logarithmic plot of data points at different tangent height. Note that negative values are not plotted, and noise is dominating at higher altitudes.

4

Linear Bayesian Model vs Regularisation Approach – Ozone

This sounds like it is the last Chapter. Is it? The hierarchical model is neither a 'Result' nor a 'Conclusion' so why is it in this Chapter? Something is very wrong with your structure. This appears to contain Modelling and Numerical Experiments - -and those should be separate Chapters. In this chapter, we use the forward model to generate some data given an underlying ground truth and then guide the reader through the process of setting up a Bayesian framework and ultimately obtaining the posterior distributions of hyper-parameters and parameters of interest, such as ozone concentration or pressure and temperature profiles. We use DAGs to visualise hierarchical and correlational structures of a Bayesian model, establish a choice of prior distributions within our Bayesian model and formulate the posterior distributions. All programming and analysis is done in Python, and the reported computation times correspond to a MacBook Pro from 2019 with a 2.4 GHz quad-core Intel Core i5 processor.

Based on the linear forward model $\boldsymbol{A}_L$, we characterise the marginal posterior for ozone and compare that to the TT approximation. Then we calculate the mean and the covariance matrix of the full posterior for ozone, which we use to generate posterior ozone samples and according to those find an affine map to approximate the non-linear forward model $\boldsymbol{A}(\boldsymbol{x}) \approx \boldsymbol{M}\boldsymbol{A}_L\boldsymbol{x}$ (see Sec. **??**). In Sec. **??**, we repeat the MTC scheme to provide a posterior distribution of ozone based on the approximate forward model and compare it to a regularisation approach and a ground truth.

In Sec **??**, we extend the hierarchical Bayesian model and MTC scheme to jointly provide posterior distributions of ozone, temperature and pressure. Here, we elaborate on some aspects of prior modelling and on our findings when using a TT to approximate the higher-dimensional marginal posterior.

4.1 Hierarchical Bayesian Framework for Ozone

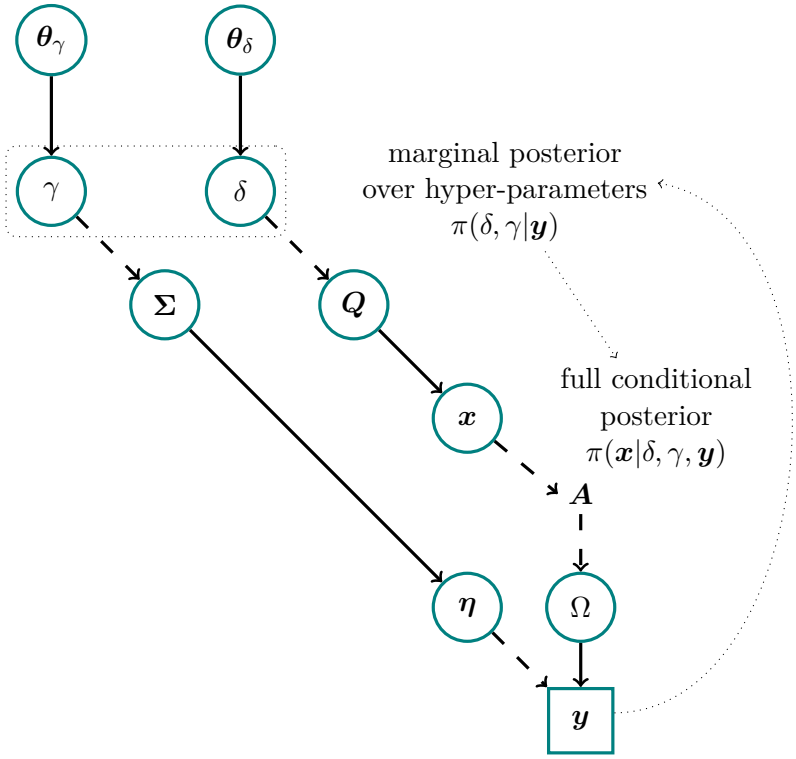


Figure 4.1: DAG for visualisation of hierarchical modelling and measuring process of ozone, including the MTC scheme. The hyper-parameter γ deterministically (dotted line) sets the noise covariance $\boldsymbol{\Sigma} = \gamma^{-1}\boldsymbol{I}$ and hence the random (solid line) noise vector $\boldsymbol{\eta} \sim \mathcal{N}(0, \gamma^{-1}\boldsymbol{I})$. The hyper-parameter δ determines (dotted line) the prior precision matrix $\boldsymbol{Q} = \delta\boldsymbol{L}$ for the normally distributed (solid line) prior $\boldsymbol{x}|\delta \sim \mathcal{N}(0, \delta\boldsymbol{L})$, where $\boldsymbol{L}$ is a graph Laplacian, see Eq. 4.2. The hyper-prior distributions (solid line) $\pi(\delta, \gamma)$ are defined by $\boldsymbol{\theta}_\gamma$ and $\boldsymbol{\theta}_\delta$. Through a linear forward model $\boldsymbol{A}$, we generate a space of all measurable noise-free data $\boldsymbol{Ax}$ from which we randomly observe a data set $\boldsymbol{y}$ including some added noise $\boldsymbol{\eta}$. Within the MTC scheme, we evaluate the marginal posterior over the hyper-parameters $\pi(\gamma, \delta|\boldsymbol{y})$ first and then the full conditional posterior $\pi(\boldsymbol{x}|\delta, \gamma, \boldsymbol{y})$. This breaks the correlation structure of $\boldsymbol{x}$ and δ and γ, and allows us to evaluate the marginal posterior independent of $\boldsymbol{x}$.

In this section, we set up the hierarchically-ordered linear-Gaussian Bayesian framework to determine the ozone posterior distribution, conditioned on ground truth temperature and pressure. Where, for now, we define the forward model matrix $\boldsymbol{A} \coloneqq \boldsymbol{A}_L$ and define the distributions of that Bayesian model, similarly to a regularisation approach, as:

$$\boldsymbol{y}|\boldsymbol{x}, \gamma, \delta \sim \mathcal{N}(\boldsymbol{A}\,\boldsymbol{x}, \gamma^{-1}\boldsymbol{I}) \tag{4.1a}$$

$$\boldsymbol{x}|\delta \sim \mathcal{N}(\boldsymbol{0}, (\delta\boldsymbol{L})^{-1}) \tag{4.1b}$$

$$\delta \sim \Gamma(\alpha_\delta, \beta_\delta) \tag{4.1c}$$

$$\gamma \sim \Gamma(\alpha_\gamma, \beta_\gamma)\,. \tag{4.1d}$$

Assuming Gaussian noise $\boldsymbol{\eta} \sim \mathcal{N}(0, \gamma^{-1}\boldsymbol{I})$, the likelihood function is a normal distribution with mean $\boldsymbol{Ax}$ and covariance matrix $\gamma^{-1}\boldsymbol{I}$. We define the normal prior-distribution

$\pi(\boldsymbol{x}|\delta)$ with zero mean and precision matrix $\delta\boldsymbol{L}$, where δ is a smoothness hyper-parameter and $\boldsymbol{L}$ is the second order discrete derivate operator (see Eq. 4.2). Here the hyper-prior distributions $\pi(\delta)$ and $\pi(\gamma)$ are gamma distributions with shape α and rate β.

We can visualise this hierarchical structure and the correlations between different hyper-parameters and parameters through a DAG, as in Fig. 4.1. The hyper-parameter γ sets the noise covariance deterministically (dotted line), but is itself statistically (solid line) defined by the hyper-prior distribution $\pi(\gamma)$. This is a gamma distribution, where $\boldsymbol{\theta}_\gamma$ determines the shape and rate of $\pi(\gamma)$. Similarly $\boldsymbol{\theta}_\delta$ defines $\pi(\delta)$, where δ accounts for smoothness of the ozone profile and sets the prior precision $\boldsymbol{Q}(\delta)$. Then $\boldsymbol{Ax}$ determines the space of all measurable noise-free data sets Ω through the linear forward model, from which we observe a data set $\boldsymbol{y}$ including some noise $\boldsymbol{\eta}$. Given that data, we "reverse the arrows" to determine the posterior distribution $\pi(\boldsymbol{x}, \boldsymbol{\theta}|\boldsymbol{y})$ over the parameter $\boldsymbol{x}$ and the hyper-parameters $\boldsymbol{\theta}$. Usually, due to underlying correlation structures, evaluating this posterior poses a signifiant challenge. The MTC scheme breaks this correlation and provides the marginal posterior $\pi(\delta, \gamma|\boldsymbol{y})$ first and then the full conditional posterior $\pi(\boldsymbol{x}|\delta, \gamma, \boldsymbol{y})$.

4.1.1 Prior Modelling

To complete the Bayesian framework, we have to define prior distributions over the hyper-parameters and parameters. Ideally, we define the prior distributions as uninformative as possible, and include functional dependencies and physical properties.

By choosing a normally distributed prior $\pi(\boldsymbol{x}|\delta)$ with zero mean and no other restrictions, it is clear that our model does not take into account that ozone values cannot be negative. As already mentioned, we set the precision matrix of that prior distribution to

$$\delta\boldsymbol{L} = \delta \begin{bmatrix} 2 & -1 & & & \\ -1 & 2 & -1 & & \\ & \ddots & \ddots & \ddots & \\ & & -1 & 2 & -1 \\ & & & -1 & 2 \end{bmatrix} \tag{4.2}$$

which is a 1-dimensional Graph Laplacian as in [62, 19] with Dirichlet boundary condition. This matrix will also act as the regulariser later in Sec. **??**. We reduce the dimension of $\boldsymbol{x}$ from 45 to 34 by discarding every second ozone VMR over a height of $\approx$ 47km. Doing that, while not changing $\boldsymbol{L}$, we effectively induce a larger correlation between points at higher altitude. We plot the corresponding prior ozone profiles according to $\boldsymbol{x} \sim \mathcal{N}(0, (\delta\boldsymbol{L})^{-1})$ in Fig. B.1.

For δ and γ we pick relatively uninformative gamma distributions so that $\gamma \sim \mathcal{T}(\boldsymbol{\theta_\gamma}) \propto \gamma^{\alpha_\gamma - 1} \exp(-\beta_\gamma \gamma)$ and $\delta \sim \mathcal{T}(\boldsymbol{\theta_\delta})$, where $\boldsymbol{\theta_\gamma} = \{\alpha_\gamma, \beta_\gamma\} = \{\alpha_\delta, \beta_\delta\} = \boldsymbol{\theta_\delta} = (1, 10^{-35})$ (see

Fig. 4.6) similar to [19]. Those gamma distributions have another advantage when using the MWG algorithm to sample from the marginal posterior distribution $\pi(\delta, \gamma|\boldsymbol{y})$, where then $\pi(\gamma|\lambda, \boldsymbol{y}) \sim \mathcal{T}(\cdot)$ is a gamma distribution with $\lambda = \delta/\gamma$, and easy to sample from.

4.2 Posterior Distribution

As explained in Sec. 2.1.1,we factorise the posterior

$$\pi(\boldsymbol{x}, \delta, \gamma|\boldsymbol{y}) \propto \pi(\boldsymbol{y}|\boldsymbol{x}, \delta, \gamma)\pi(\boldsymbol{x}, \delta, \gamma) \tag{4.3}$$

into

$$\pi(\boldsymbol{x}, \delta, \gamma|\boldsymbol{y}) = \pi(\boldsymbol{x}|\delta, \gamma, \boldsymbol{y})\pi(\delta, \gamma|\boldsymbol{y}) \tag{4.4}$$

the marginal posterior $\pi(\delta, \gamma|\boldsymbol{y})$ and full conditional posterior $\pi(\boldsymbol{x}|\delta, \gamma, \boldsymbol{y})$ (see Eq. 2.7). As discussed in Sec. 2.1.1, for the linear-Gaussian case, $\boldsymbol{x}$ cancels in the marginal posterior over the hyper-parameters. Following the MTC scheme, we characterise the marginal posterior first and then the full conditional posterior.

4.2.1 Marginal posterior

Sample from marginal posterior – Metropolis within Gibbs

If $\boldsymbol{\theta} \in \mathbb{R}^2$, really? Cut the opaque speak and say it has two components. we use a Metropolis-within-Gibbs (MWG) sampler as described in [19] to sample from $\pi(\boldsymbol{\theta}|\boldsymbol{y})$. With $\boldsymbol{\theta} = (\theta_1, \theta_2)$, we perform a Metropolis step MwG is the standard abbreviation. Anyway, we could use a Gibbs sampler if there are 5 components. What is this trying to say? in the θ_1 direction and a Gibbs step in the θ_2 direction. Ergodicity for this approach is proven in [45].

The MWG algorithm starts at the 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 Metropolis step and a special case of the Metropolis-Hastings algorithm [45]. We accept and set $\theta_1^{(t)} = \theta_1$ with the acceptance probability

$$\alpha(\theta_1|\theta_1^{(t-1)}) = \min\left\{1, \frac{\pi(\theta_1|\theta_2^{(t-1)}, \boldsymbol{y})\, \cancel{q(\theta_1^{(t-1)}|\theta_1)}}{\pi(\theta_1^{(t-1)}|\theta_2^{(t-1)}, \boldsymbol{y})\, \cancel{q(\theta_1|\theta_1^{(t-1)})}}\right\} \tag{4.5}$$

or reject , otherwise reject and keep $\theta_1^{(t)} = \theta_1^{(t-1)}$, which we do by comparing α to a uniform random number $u \sim \mathcal{U}(0, 1)$. If you want to say you calculate to accept by this method, it's a separate sentence, Mr wandering sentence man.

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 this seems a very specific algorithm

for a Chapter that you said would be general. You must have a specific distribution inmind, or are you saying that this is a thing that you could do? I have no idea why you are writing this. 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 $N_{\text{burn-in}}$ initial samples after a burn-in period, resulting in a Markov chain of length $N = N' - N_{\text{burn-in}}$. what is N'? Have you defined any of these things? What is this all doing here?

Algorithm 3: Metropolis within Gibbs

1: Initialise two-dimensional vector $\boldsymbol{\theta}^{(0)} = (\theta_1^{(0)}, \theta_2^{(0)})$
2: **for** $k = 1, \ldots, N'$ **do**
3: Propose $\theta_1 \sim q(\cdot|\theta_1^{(t-1)}) = q(\theta_1^{(t-1)}|\cdot)$
4: Compute

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

5: Draw $u \sim \mathcal{U}(0,1)$
6: **if** $\alpha \geq u$ **then**
7: Accept and set $\theta_1^{(t)} = \theta_1$
8: **else**
9: Reject and keep $\theta_1^{(t)} = \theta_1^{(t-1)}$
10: **end if**
11: Draw $\theta_2^{(t)} \sim \pi(\cdot|\theta_1^{(t)}, \boldsymbol{y})$
12: **end for**
13: Output: $\boldsymbol{\theta}^{(0)}, \ldots, \boldsymbol{\theta}^{(k)}, \ldots, \boldsymbol{\theta}^{(N)} \sim \pi(\boldsymbol{\theta}|\boldsymbol{y})$

Consequently, for the hierarchical model specified in Eq. 4.1, the marginal posterior distribution over the hyper-parameters is given by

$$\pi(\lambda, \gamma|\boldsymbol{y}) \propto \lambda^{n/2+\alpha_\delta-1}\gamma^{m/2+\alpha_\delta+\alpha_\gamma-1} \exp\left\{-\frac{1}{2}g(\lambda) - \frac{\gamma}{2}f(\lambda) - \beta_\delta\lambda\gamma - \beta_\gamma\gamma\right\}, \tag{4.6}$$

with the introduced regularisation parameter $\lambda = \delta/\gamma$, and

$$f(\lambda) = \boldsymbol{y}^T\boldsymbol{y} - (\boldsymbol{A}^T\boldsymbol{y})^T(\boldsymbol{A}^T\boldsymbol{A} + \lambda\boldsymbol{L})^{-1}(\boldsymbol{A}^T\boldsymbol{y})\,, \tag{4.7a}$$

$$\text{and } g(\lambda) = \log\det(\boldsymbol{A}^T\boldsymbol{A} + \lambda\boldsymbol{L})\,. \tag{4.7b}$$

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 $\mathrm{d}\delta/\mathrm{d}\lambda = \gamma$. For each evaluation of the marginal posterior most of the computational effort lies in the calculation of $f(\lambda)$ and $g(\lambda)$, which we evaluate using the Cholesky decomposition $\boldsymbol{A}^T\boldsymbol{A} + \lambda\boldsymbol{L} = \boldsymbol{C}_\lambda\boldsymbol{C}_\lambda^T$. More specifically we solve for $(\boldsymbol{A}^T\boldsymbol{A} + \lambda\boldsymbol{L})^{-1}(\boldsymbol{A}^T\boldsymbol{y})$ via `scipy.linalg.cho_solve` and calculate the $g(\lambda) =$

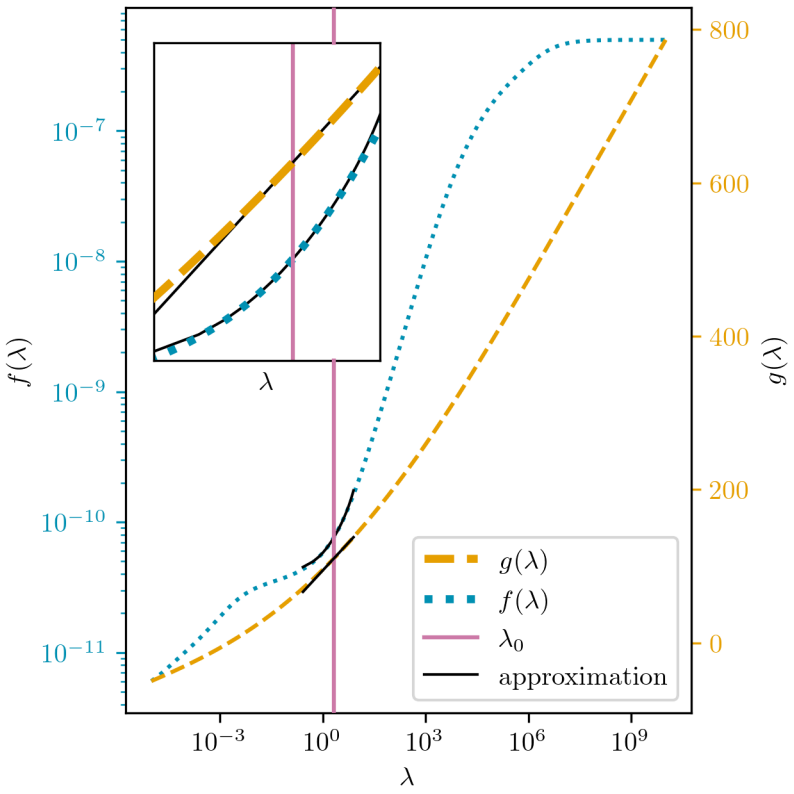


Figure 4.2: Functions $f(\lambda)$ and $g(\lambda)$ from the marginal posterior in Eq. 4.6 for a wide range of $\lambda = \delta/\gamma$. We plot the approximations (see Eq. 4.10a and Eq. 4.10b) in black around the mode of the marginal posterior (vertical line) for the sampling range of λ within the MWG algorithm.

$2\sum \log \operatorname{diag}(\boldsymbol{C}_\lambda)$. In Fig. 4.2 we see that $f(\lambda)$ and $g(\lambda)$ are well behaved within the region of interest. Because of this, we approximate $f(\lambda) \approx \tilde{f}(\lambda)$ with a Taylor series and $\tilde{g}(\lambda) \approx g(\lambda)$ with a linear approximation in log-space around the mode λ_0 of $\pi(\lambda, \gamma|\boldsymbol{y})$. The approximations are implicitly given by

$$f^{(r)}(\lambda_0) = (-1)^{r+1}(\boldsymbol{A}^T\boldsymbol{y})^T(\boldsymbol{B}_0^{-1}\boldsymbol{L})^r\boldsymbol{B}_0^{-1}\boldsymbol{A}_L^T\boldsymbol{y} \tag{4.8}$$

$$\text{and } \log\tilde{g}(\lambda) = \log g(\lambda_0) + (\log\lambda - \log\lambda_0)\frac{\log g(\lambda_{\max}) - \log g(\lambda_0)}{\log\lambda_{\max} - \log\lambda_0} \tag{4.9}$$

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

$$\tilde{f}(\lambda) = \sum_{r=0}^{2} f^{(r)}(\lambda_0)(\lambda - \lambda_0)^r\,, \tag{4.10a}$$

$$\text{and } \tilde{g}(\lambda) = \exp\log\tilde{g}(\lambda)\,, \tag{4.10b}$$

in Fig. 4.2 and elaborate on the approximation errors in the section below. Usually a Taylor series includes a factor $(r!)^{-1}$, which in this case cancels in $f^{(r)}(\lambda_0)$, and $g(\lambda)$ can be approximated with a Taylor series as well (see [19]).

We approximate $f(\lambda)$ and $g(\lambda)$ around the mode (λ_0, γ_0) of $\pi(\lambda, \gamma|\boldsymbol{y})$ provided by the `scipy.optimize.fmin` function, with a limit of 25 function evaluations. Then we approximate $f(\lambda)$ with a 2-nd order Taylor series and $g(\lambda)$ with a linear approximation in the log-space, where for the approximation in $g(\lambda)$ we set $\lambda_{\max}$ to the maximum value of λ on the TT grid (see next section).

We initialised the MWG algorithm at the mode $(\lambda^{(0)}, \gamma^{(0)}) = (\lambda_0, \gamma_0)$ and take $N = 10000$ plus $N_{\text{burn-in}} = 100$ steps in ≈ 0.5s. The standard deviation of the normal proposal distribution is empirically set to $\sigma_\lambda = 0.8\lambda_0$, so that the acceptance rate is ≈ 0.5 as suggested in [43]. More specifically, we implement a Metropolis random walk on

$$\pi(\lambda|\gamma, \boldsymbol{y}) \propto \lambda^{n/2+\alpha_\delta-1} \exp\Big\{-\frac{1}{2}g(\lambda) - \frac{\gamma}{2}f(\lambda) - \beta_\delta\gamma\lambda\Big\}. \tag{4.11}$$

In doing so, we accept $\lambda^{k+1} = \lambda'$ or reject $\lambda^{k+1} = \lambda^k$ a proposal $\lambda' \sim \mathcal{N}(0, \sigma_\lambda^2)$ according to the acceptance ratio

$$\alpha(\lambda'|\lambda^k) = \min\left\{1, \frac{\pi(\lambda'|\gamma^{(k)}, \boldsymbol{y})}{\pi(\lambda^{(k)}|\gamma^{(k)}, \boldsymbol{y})}\right\}. \tag{4.12}$$

In practice, we calculate the acceptance ratio in log space, so that

$$\log\left\{\frac{\pi(\lambda'|\gamma^{(k)}, \boldsymbol{y})}{\pi(\lambda^{(k)}|\gamma^{(k)}, \boldsymbol{y})}\right\} = \log\{\pi(\lambda'|\gamma^{(k)}, \boldsymbol{y})\} - \log\{\pi(\lambda^{(k)}|\gamma^{(k)}, \boldsymbol{y})\} \tag{4.13}$$

$$= \frac{n}{2}(\log\{\lambda'\} - \log\{\lambda^{(k)}\}) + \frac{1}{2}\Delta g + \frac{\gamma^{(k)}}{2}\Delta f + \beta_\delta\gamma^{(k)}\Delta\lambda\,, \tag{4.14}$$

where $\Delta\lambda = \lambda' - \lambda^{(k)}$ and $\Delta f \approx \tilde{f}(\lambda') - \tilde{f}(\lambda^{(k)}) = \sum_1^2 f^{(r)}(\lambda_0)[(\Delta\lambda')^r - (\Delta\lambda^{(k)})^r]$, with $\Delta\lambda' = \lambda' - \lambda_0$ and $\Delta\lambda^{(k)} = \lambda^{(k)} - \lambda_0$. Similarly we approximate $\Delta g \approx \tilde{g}(\lambda') - \tilde{g}(\lambda^{(k)})$. Lastly, we do a Gibbs step on

$$\gamma^{(k+1)}|\lambda^{(k+1)}, \boldsymbol{y} \sim \Gamma\Big(\frac{m}{2} + \alpha_\delta + \alpha_\gamma, \frac{1}{2}f(\lambda^{(k+1)}) + \beta_\gamma + \beta_\delta\lambda^{(k+1)}\Big) \tag{4.15}$$

to generate marginal posterior samples $(\lambda, \gamma)^{(1)}, \ldots, (\lambda, \gamma)^{(N)} \sim \pi(\lambda, \gamma|\boldsymbol{y})$, which we plot in Fig. 4.4 as well as the trace of the MWG to show ergodicity. We calculate the IACT with the Python implementation of [64] provided by [28], which gives us $\tau_{\text{int},\gamma} \approx 4.4 \pm 0.2$ and $\tau_{\text{int},\lambda} = 10.4 \pm 1.0$ (see Fig. 4.3 and Fig. B.2).

TT approximation of marginal posterior

Alternatively, we can utilise a TT approximation of the square root of the marginal posterior over a predefined grid and calculate the marginals $\pi(\gamma|\boldsymbol{y})$ and $\pi(\lambda|\boldsymbol{y})$ (see Sec. 2.2.1).

We predefined the univariate grid with $n = 20$ grid points (see Fig. 4.8) over $\gamma = [0.25 \times 10^{15}, 6 \times 10^{15}]$ and $\lambda = [1, 5000]$. We introduce a "normalisation constant" $c = -340/2$ to avoid underflow so that the values of $\sqrt{\pi(\lambda, \gamma|\boldsymbol{y})} = \exp\{0.5 \log \pi(\lambda, \gamma|\boldsymbol{y}) + c\}$ are within

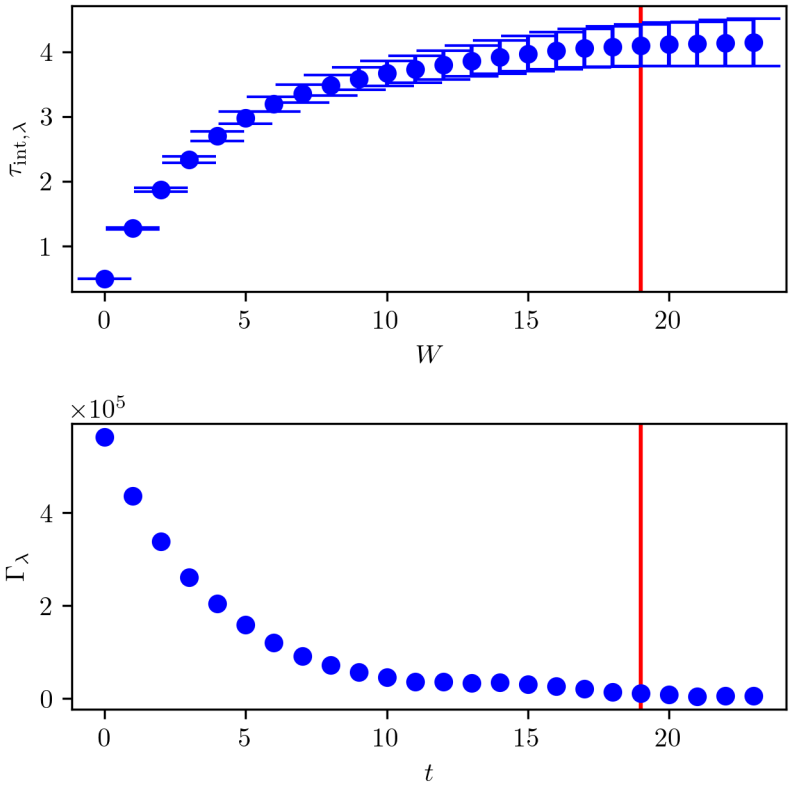


Figure 4.3: Provided by [28], the IACT $\tau_{\text{int},\lambda}$ at summation windows W as well as the estimated autocorrelation function Γ_λ at lag t of the samples $\lambda \sim \pi(\cdot|\boldsymbol{y})$.

computer precision. Then we initialise the `tt.cross.rectcross.rect_cross.cross` function based on the TT cross algorithm in [40, 12] from the Python package `ttpy` [39] with a random tensor. The number of ranks is set to constant $r = 4$, we do 1 sweep with $2n_{\text{sweeps}}2nr = 400$ function evaluations and obtain a TT approximation of $\pi(\lambda, \gamma|\boldsymbol{y})$ in about 0.02s. Ironically, this the same number of functions evaluations to approximate a 20×20 point grid. The TT format is especially advantageous for larger grid sizes and higher dimensional parameter spaces. To compute the marginals $\pi(\lambda|\boldsymbol{y})$ and $\pi(\gamma|\boldsymbol{y})$ we set the TT error $\xi = 1/\boldsymbol{\lambda}(\mathcal{X})$ and $\boldsymbol{\lambda}(x) = 1$, so that for Cartesian basis $\boldsymbol{M}_k = \text{diag}(\boldsymbol{\lambda}_k(\mathcal{X}_k))$ (see Eq. 2.28). We calculate the coefficient tensor $\boldsymbol{B}$ and $\boldsymbol{R}_{\text{pre}}$ as in Prop. 1 and Prop. 2 (see Sec. 2.2.1).

We plot the TT approximation as a colour map on top of the obtained samples in Fig. 4.4. The relative RMS TT approximation error over the whole grid is $\approx 3\%$ and similar to the propagation error in $\pi(\lambda, \gamma|\boldsymbol{y})$ due to the approximations of $f(\lambda)$ and $g(\lambda)$ (see further up).

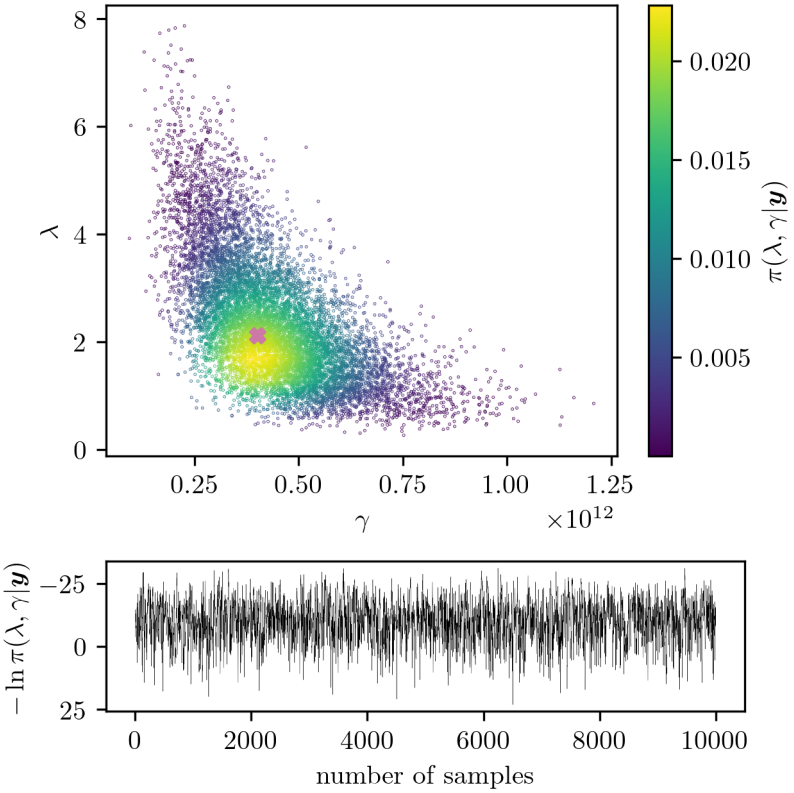


Figure 4.4: Samples from the marginal posterior colour-coded using the TT approximation of $\pi(\lambda, \gamma|\boldsymbol{y})$. The mode of (λ_0, γ_0) of $\pi(\lambda, \gamma|\boldsymbol{y})$ is marked with the pink cross. To show ergodicity, we plot the trace of the samples of the MWG algorithm.

Error due to approximation of f and g

To assess the approximation error, we lay a 100-point grid over the sampling region in each dimension and compare the approximations of $f(\lambda)$, $g(\lambda)$ and $\pi(\lambda, \gamma|\boldsymbol{y})$ with their true function values.

Compared to a 1-st, 3-rd or 4-th order Taylor approximation, we find that the 2-nd order Taylor approximation of $f(\lambda)$ gives the smallest relative RMS error of $\approx 10\%$ in the sampling region of λ. Additionally, we find that a linear approximation of $g(\lambda)$ is sufficient with relative RMS $< 0.1\%$.

These errors then propagate into the marginal posterior $\pi(\lambda, \gamma|\boldsymbol{y})$ so that the relative RMS error is $\approx 3\%$ over the whole grid. When sampling, we evaluate the acceptance ratio in the log-space, so we report a relative RMS error of $< 0.01\%$ for $\log \pi(\lambda|\gamma, \boldsymbol{y})$, with a maximum relative pointwise error of $< 0.5\%$. We consider this good enough.

Using these approximations, we employ a Metropolis within Gibbs (MWG) sampler (see Alg. Box 3) to characterise $\pi(\lambda, \gamma|\boldsymbol{y})$ (see Sec. 4.2.1).

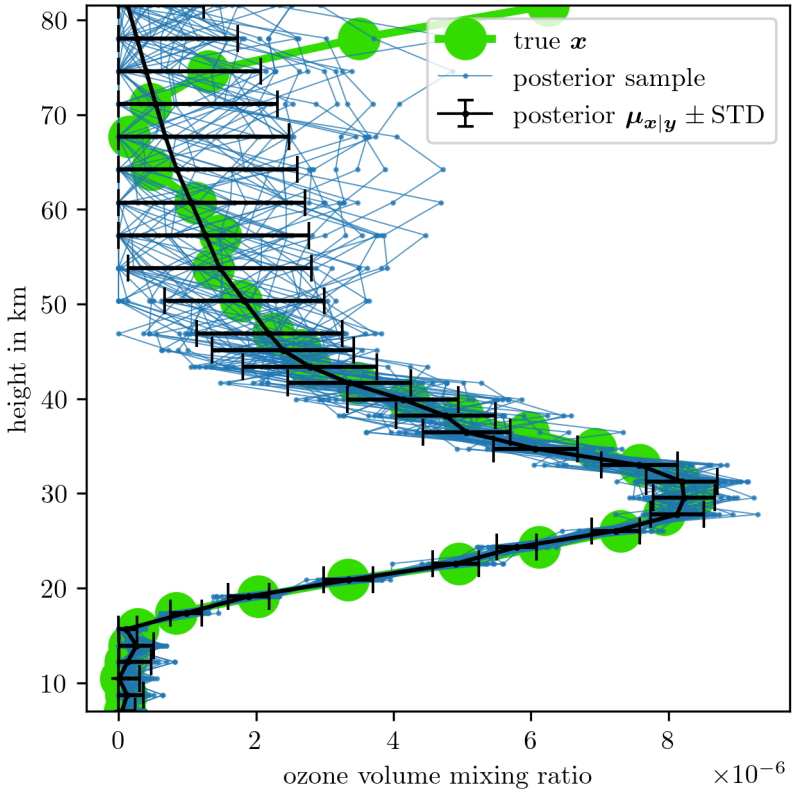


Figure 4.5: Ozone samples from the full posterior distribution $\pi(\boldsymbol{x}|\boldsymbol{y})$ after characterising full posterior mean and covariance by weighted expectations over the marginal posterior $\pi(\lambda, \gamma|\boldsymbol{y})$ based on the linear forward map $\boldsymbol{A}_L$. We set negative ozone VMR values to zero.

4.2.2 Full posterior ozone mean and variance

Finally, we evaluate the normally distributed full conditional posterior distribution

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

as in Eq. 2.15, with $\lambda = \delta/\gamma$. In this thesis, we compute the mean

$$\mu_{\boldsymbol{x}|\boldsymbol{y}} = \int \boldsymbol{x}_\lambda \pi(\lambda|\boldsymbol{y})\, \mathrm{d}\lambda \approx \sum \boldsymbol{x}_{\lambda_i} \pi(\lambda_i|\boldsymbol{y})\,, \tag{4.17}$$

and covariance

$$\Sigma_{\boldsymbol{x}|\boldsymbol{y}} = \int \gamma^{-1}\pi(\gamma|\boldsymbol{y})\, \mathrm{d}\gamma \int \boldsymbol{B}_\lambda^{-1}\, \pi(\lambda|\boldsymbol{y})\, \mathrm{d}\lambda \approx \sum {\gamma_i}^{-1} \pi(\gamma_i|\boldsymbol{y}) \sum \boldsymbol{B}_{\lambda_i}^{-1} \pi(\lambda_i|\boldsymbol{y}) \tag{4.18}$$

of $\pi(\boldsymbol{x}|\boldsymbol{y})$ as weighted expectations over the marginal posterior $\pi(\lambda, \gamma|\boldsymbol{y})$ by quadrature [11, Sec. 2.1] with $\sum \pi(\lambda_i|\boldsymbol{y}) = \sum \pi(\gamma_i|\boldsymbol{y}) = 1$. The weights $\pi(\lambda_i|\boldsymbol{y})$ and $\pi(\gamma_i|\boldsymbol{y})$ are either given by the TT approximation or by the bars of the sample-based histograms. More precisely, the heights of the sample-based histogram bars act as quadrature weights, where λ_i is defined at the centre of each bar. We use Cholesky decomposition of

$\boldsymbol{B}_\lambda = \boldsymbol{A}^T\boldsymbol{A} + \lambda\boldsymbol{L}$ to invert $\boldsymbol{B}_\lambda$ and to calculate $\boldsymbol{x}_\lambda = (\boldsymbol{A}^T\boldsymbol{A} + \lambda\boldsymbol{L})^{-1}\boldsymbol{A}^T\boldsymbol{y}$ both via `scipy.linalg.cho_solve`. It is sufficient to evaluate $\boldsymbol{x}_\lambda$ and invert $\boldsymbol{B}_\lambda$ 20 times to obtain mean and covariance values of $\pi(\boldsymbol{x}|\boldsymbol{y})$ within a reasonable error (see Fig. 4.8). We need roughly 0.025s to calculate the full posterior mean and variance including finding the mode of $\pi(\lambda, \gamma|\boldsymbol{y})$, running the TT `cross` and calculating the marginals. If we use the MWG sampler we need ≈ 0.5s for the same results, so most computational effort lays within the sampling procedure and the time to calculate full posterior mean and variance is negligible. We plot posterior samples of $\pi(\boldsymbol{x}|\boldsymbol{y})$ in Fig. 4.5 and set negative ozone values to zero, which we observe in almost every sample. The fact that we have to deal with negative ozone values is due to the poor prior choice in $\pi(\boldsymbol{x}|\delta)$. This indicates that we should use a different, more physically based prior or model a parametrised ozone profile. Note that the posterior samples do not capture the second ozone peak at around 80km.

If calculating the variance is too costly, the RTO method (see Sec. 6.3.1) may be a feasible alternative to draw a sample from Eq. 4.16.

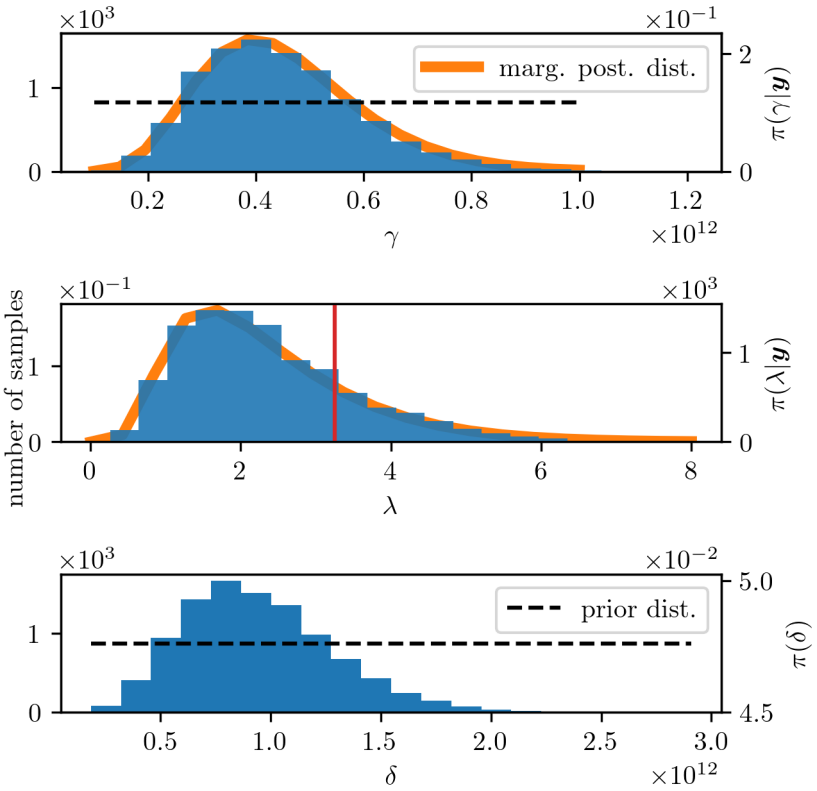


Figure 4.6: The TT approximation of the marginal posterior in orange and the samples as a histogram, as well as the prior distribution with a dotted line. We sample λ and γ using the MWG algorithm and then calculate δ for every sample of the marginal posterior. The regularised parameter corresponding to the best regularised solution (see Fig. 4.10 and Fig. 4.9) is marked with the red vertical line. We mark the ground truth noise precision with the back vertical line.

The marginal posterior is defined as in Eq. 4.6, but with the approximated forward model. We initialise the MWG at and approximate $f(\lambda)$ and $g(\lambda)$ around the mode of $\pi(\lambda, \gamma|\boldsymbol{y})$ (see Eq. 4.10a and Eq. 4.10b), and take $N = 10000$ plus $N_{\text{burn-in}} = 100$ steps. The IACTs provided by [28] are $\tau_{\text{int},\gamma} \approx 4.0 \pm 0.2$ and $\tau_{\text{int},\lambda} = 8.6 \pm 0.8$ (see Fig. B.4 and Fig. B.3) and similar to the previously calculated values. We plot the samples in Fig. 4.6 as well as TT approximation of the marginal posterior using 400 function evaluations (same grid; same number of ranks; see Sec. 4.2.1). The approximation error of the TT is $\approx 3\%$ and similar to the error due approximations of $f(\lambda)$ and $g(\lambda)$.

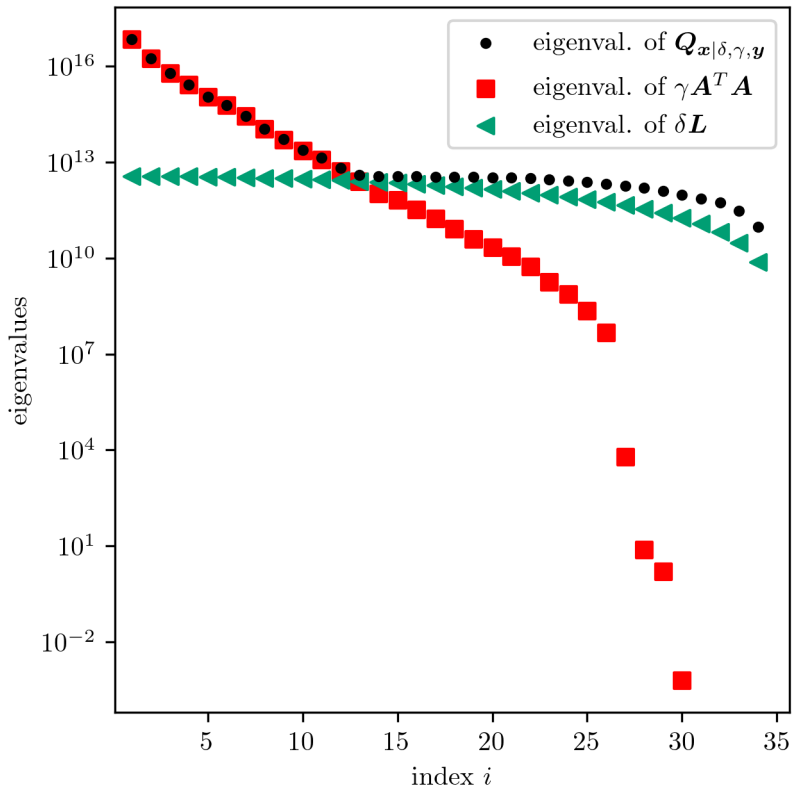


Figure 4.7: Eigenvalues of the precision matrix of $\boldsymbol{Q}_{\boldsymbol{x}|\delta,\gamma,\boldsymbol{y}} = \gamma \boldsymbol{A}^T \boldsymbol{A} + \delta \boldsymbol{L}$ of the full posterior distribution $\pi(\boldsymbol{x}|\delta, \gamma, \boldsymbol{y})$ for ozone. We see that large eigenvalues of $\gamma \boldsymbol{A}^T \boldsymbol{A}$ and $\delta \boldsymbol{L}$ are rather unaffected by the prior compared to small eigenvalues. The eigenbasis may differ.

Eigenvalues full conditional posterior covariance We can visualise how the posterior samples are affected through the prior $\delta \boldsymbol{L}$ and the forward model $\gamma \boldsymbol{A}^T \boldsymbol{A}$ by comparing their eigenvalues to the eigenvalues of the precision matrix $\boldsymbol{Q}_{\boldsymbol{x}|\delta,\gamma,\boldsymbol{y}} = \gamma \boldsymbol{A}^T \boldsymbol{A} + \delta \boldsymbol{L}$ for a random $\delta, \gamma \sim \pi(\delta, \gamma|\boldsymbol{y})$. We order the eigenvalues in size and plot those in Fig. 4.7. We observe that the larger eigenvalues of $\boldsymbol{Q}_{\boldsymbol{x}|\delta,\gamma,\boldsymbol{y}}$ are very much the same as the larger eigenvalues of $\gamma \boldsymbol{A}^T \boldsymbol{A}$. Once the eigenvalues of $\gamma \boldsymbol{A}^T \boldsymbol{A}$ are significantly smaller than the eigenvalues of $\boldsymbol{Q}_{\boldsymbol{x}|\delta,\gamma,\boldsymbol{y}}$ the structure of the eigenvalues is dominated by the eigenvalues

of $\delta \boldsymbol{L}$. The largest 10 eigenvalues of $\boldsymbol{Q}_{\boldsymbol{x}|\delta,\gamma,\boldsymbol{y}}$ include ozone profile structure at lower altitudes, where the other eigenvector mainly represent structures at higher altitudes (see Fig. B.5 and Fig. B.5). Note that the eigenvalues of each matrix may correspond to different eigenvectors even if the eigenvalues of two matrices are the same.

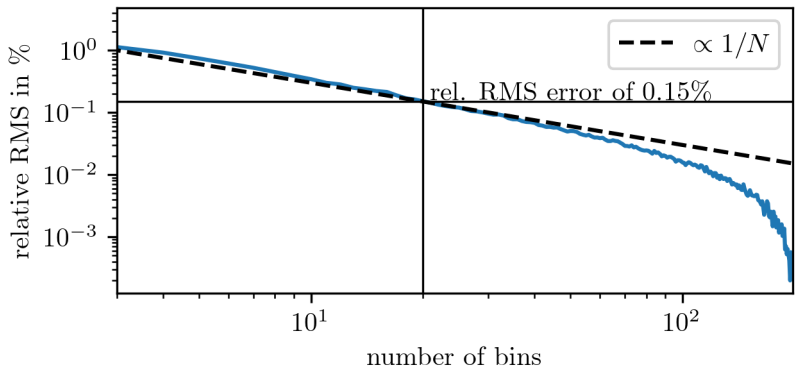


Figure 4.8: Relative RMS error of $\boldsymbol{\mu}_{\boldsymbol{x}|\boldsymbol{y}}$ and covariance $\boldsymbol{\Sigma}_{\boldsymbol{x}|\boldsymbol{y}}$ calculated by the weighted expectations and compared to a "ground truth" given by weighted expectations over 200 bins.

Errors of full posterior mean and covariance In Fig. 4.8, we plot the relative RMS error for the mean $\boldsymbol{\mu}_{\boldsymbol{x}|\boldsymbol{y}}$ and covariance $\boldsymbol{\Sigma}_{\boldsymbol{x}|\boldsymbol{y}}$ of $\pi(\boldsymbol{x}|\boldsymbol{y})$. We obtain those results by calculating the weighted expectation over normalised histograms of $\pi(\lambda, \gamma|\boldsymbol{y})$, where we vary the number of bins and compare to a solution calculated from a histogram with 200 bins. The relative error behaves roughly proportional to $1/N$, and we consider a relative error less than 0.5% good enough, which we easily meet at 20 bins. This sets the TT grid size and the number of evaluations of $\boldsymbol{x}_\lambda$ in Eq. 4.17 and $(\gamma \boldsymbol{B}_\lambda)^{-1}$ in Eq. 4.18.

4.3 Solution by Regularisation

Since we claim that the Bayesian approach is superior to regularisation methods, we compare the MTC method to a Tikhonov regularisation, see Sec. **??** and [19]. This is most similar to our chosen linear-Gaussian Bayesian framework.

The regularised solution is defined as in [24, 19]

$$\boldsymbol{x}_\lambda = \arg\min_{\boldsymbol{x}} \|\boldsymbol{A}\boldsymbol{x} - \boldsymbol{y}\|_{L^2}^2 + \lambda \boldsymbol{x}^T \boldsymbol{L}\boldsymbol{x}\,, \tag{4.19}$$

with the regularisation parameter λ, and is typically calculated by solving the normal equations

$$\boldsymbol{x}_\lambda = (\boldsymbol{A}^T\boldsymbol{A} + \lambda \boldsymbol{L})^{-1}\boldsymbol{A}^T\boldsymbol{y} \tag{4.20}$$

(see Sec. **??**). To find the best regularised solution, we use the L-curve method, and follow [27]. Within this method we compute $\boldsymbol{x}_\lambda$, for 200 different λ values in between 10^0 to 10^6 and plot the solution semi norm $\sqrt{\boldsymbol{x}_\lambda^T \mathbf{L}\boldsymbol{x}_\lambda}$ against the data misfit norm $\|\boldsymbol{A}\boldsymbol{x}_\lambda - \boldsymbol{x}\|_{L^2}$ (see Figure 4.9). The regularised solution corresponds to the "corner" of the L-curve at the point of maximum curvature provided by the kneedle algorithm [51] using the function `kneed.KneeLocator` in ≈ 0.015s, which is slightly faster then the TT approach to obtain full posterior mean and covariance. We plot the corresponding regularisation parameter in Fig. 4.6, which can vary significantly compared to the λ-samples of the marginal posterior.

The regularised solution in Fig. 4.10 is very similar to the posterior mean. It is pretty clear that the regularised solution accounts for only one possible solution and does not provide uncertainties. The regularised solution is not similar to the samples drawn from the posterior $\pi(\boldsymbol{x}|\boldsymbol{y})$ (see Fig. 4.5). The samples of $\pi(\boldsymbol{x}|\boldsymbol{y})$ plotted in Fig. 4.9 lie above the L-Curve, whereas the posterior mean and the regularised solution are on the L-Curve. This does make sense, if one thinks about the mean as the (smooth) average over less-smooth samples and the regularised solution as an extremely smooth ozone profile (see Lagrange multiplier in Sec. **??**). In contrast, the samples are less regularised and hence lie above the L-Curve, but have a similar data misfit norm, and as already mentioned, are all feasible solutions to the data. Neither the regularisation solution nor the posterior ozone profiles capture the second ozone peak of the ground truth at high altitudes.

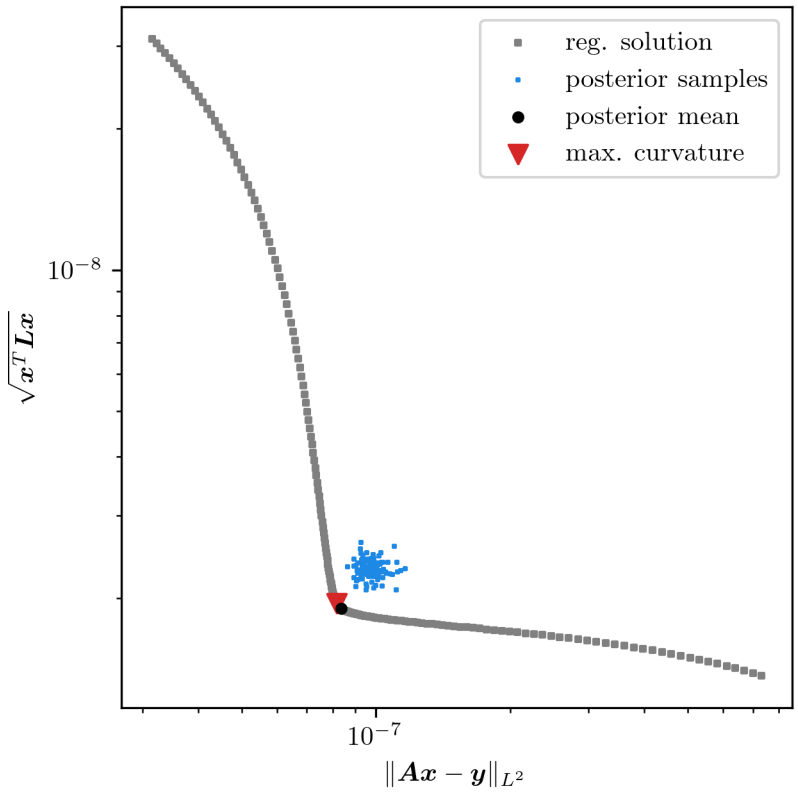


Figure 4.9: L-Curve of regularised semi norm $\sqrt{\boldsymbol{x}^T\boldsymbol{L}\boldsymbol{x}}$ against the data misfit norm $\|\boldsymbol{A}\boldsymbol{x}_\lambda - \boldsymbol{x}\|_{L^2}$ for different λ values, where $\boldsymbol{x}_\lambda$ is calculated as in Eq. 4.20. The best regularised solution is at the point of maximum curvature (pink triangle). Additionally, we calculate the data misfit norm and the regularised norm for the mean (black circle) and samples (blue squares) of the full posterior of ozone.

As mentioned in the introduction, the currently most used method to analyse data in atmospheric physics is regularisation-based. Since we want to show that Bayesian methods provide more information than regularisation at a similar computational cost, we choose a regularisation approach closest to the linear-Gaussian Bayesian framework in Sec. 4.1. Don't say that you picked an example to give a particular result. The implication is that you could pick another example to get a different result. Actually, this model is chosen to be the closest equivalent in a Bayesian model.

The Tikhonov you seem you have misunderstood naming, Tikhonov is only T=I, in your notation. regularisation approach provides one solution $\boldsymbol{x}_\lambda$ that minimises both the data misfit norm

$$\|\boldsymbol{y} - \boldsymbol{A}\boldsymbol{x}\|_{L^2} \tag{4.21}$$

and a regularisation semi-norm why is lambda in the semi-norm. If it is there, this is a family of semi-norms.

$$\lambda \|\boldsymbol{T}\boldsymbol{x}\|_{L^2} \ , \tag{4.22}$$

for a given regularisation parameter $\lambda > 0$ as described in [19], with a linear forward model matrix $\boldsymbol{A}$, the data $\boldsymbol{y}$, a regularisation operator $\boldsymbol{T}$. Next part is your sentence creep. Make two clear sentences, not one lengthy, going on too long, garbled while saying something else and the queen likes to read it on Sundays, when there is no rain or a poodle has too many walks, and also the evening news to be interesting. The regularisation parameter weights the semi-norm and penalises $\boldsymbol{x}$ according to that. If λ is large, then the effect of the data on the solution $\boldsymbol{x}_\lambda$ is small or negligible and dominated by the regulariser. what does that mean – the solution is essentially equal to the noisy data, or what? If you are going to say anything, you need to be specific. Currently this says essentially nothing. A person who knows what is going on does not need to read this sentence (or section) while a person who does not know will still have no idea after reading this. If λ is small, the solution $\boldsymbol{x}_\lambda$ will be dominated by the noisy data, resulting in an overfitted (noisy) $\boldsymbol{x}_\lambda$. We refer to [25] and [57] for a more comprehensive analysis on the effects of the regularisation parameter to the solution, e.g. due to small singular values of the forward model.

For a fixed λ, the regularised solution is given by

$$\boldsymbol{x}_\lambda = \arg\min_{\boldsymbol{x}} \|\boldsymbol{y} - \boldsymbol{A}\boldsymbol{x}\|_{L^2}^2 + \lambda \|\boldsymbol{T}\boldsymbol{x}\|_{L^2}^2 \tag{4.23}$$

is obtained which can be calculated by ... by taking the derivative with respect to $\boldsymbol{x}$:

$$\nabla_{\boldsymbol{x}} \left\{ (\boldsymbol{y} - \boldsymbol{A}\boldsymbol{x})^T (\boldsymbol{y} - \boldsymbol{A}\boldsymbol{x}) + \lambda \boldsymbol{x}^T \boldsymbol{T}^T \boldsymbol{T}\boldsymbol{x} \right\} = 0 \tag{4.24}$$

$$\iff \quad \nabla_{\boldsymbol{x}} \left\{ \boldsymbol{y}^T\boldsymbol{y} + \boldsymbol{x}^T \boldsymbol{A}^T \boldsymbol{A}\boldsymbol{x} - 2\boldsymbol{y}^T \boldsymbol{A}\boldsymbol{x} + \lambda \boldsymbol{x}^T \boldsymbol{T}^T \boldsymbol{T}\boldsymbol{x} \right\} = 0 \tag{4.25}$$

$$\iff \quad 2\boldsymbol{A}^T \boldsymbol{A}\boldsymbol{x} - 2\boldsymbol{A}^T \boldsymbol{y} + 2\lambda \boldsymbol{T}^T \boldsymbol{T}\boldsymbol{x} = 0, \tag{4.26}$$

also known as the "regularised normal equation" $\boldsymbol{A}^T\boldsymbol{y} = \boldsymbol{A}^T\boldsymbol{A}\boldsymbol{x} + \lambda\boldsymbol{T}^T\boldsymbol{T}\boldsymbol{x}$ [26]. no, these are the normal equations – the equatios for gradient equals zero, but 'reglarised normal equations' makes no sense. They are not a regularised version of the normal equations. Solving this equation yields the regularised solution

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

where we define $\boldsymbol{L} \coloneqq \boldsymbol{T}^T\boldsymbol{T}$, which typically represents a discrete matrix approximation of a differential operator choice [57]. more sentence creep. Yoiu need to define L separately, what is a "differential operator choice"? This reads very strangely. For example

$$\boldsymbol{T} = \frac{1}{h}\begin{bmatrix} -1 & 1 & & & \\ 0 & -1 & 1 & & \\ & \ddots & \ddots & \ddots & \\ & & 0 & -1 & 1 \\ & & & 0 & -1 \end{bmatrix}, \tag{4.28}$$

is the first order derivative with equal spacing h as in [57] then no, it's a first order forward differnece operator, that approximates a derivative operator.

$$\boldsymbol{L} = \frac{1}{h^2}\begin{bmatrix} 1 & -1 & & & \\ -1 & 2 & -1 & & \\ & \ddots & \ddots & \ddots & \\ & & -1 & 2 & -1 \\ & & & -1 & 2 \end{bmatrix}, \tag{4.29}$$

is the second order derivative with Neumann boundary conditions, see [62]. no it's not, it's a second-difference operator, that is a discrete approximation to the second derivative (no 'order')

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 semi-norm. The optimal value of λ corresponds to the point of maximum curvature of the so-called L-curve [27], as in Fig. 4.9 where we plot the data misfit norm versus the regularisation semi-norm. you use of the tem 'optimal' is offensive – because it is a tautology that you define 'optimal' to mean that it is given by the L-curve. So say it is optimal because it is given by th eL-curve is saying nothing, other than defining what you mean by an 'optimal regularisation parameter'. If that's what you want to say, then say it. otherwise this sentence is content free. never write 'so called' again in your life.

Alternatively one can think about regularisation as a introducing Lagrange multiplier $\mathcal{L}(\boldsymbol{x}, \lambda) \coloneqq \lambda\sqrt{\boldsymbol{x}^T\boldsymbol{L}\boldsymbol{x}} + \|\boldsymbol{y} - \boldsymbol{A}\boldsymbol{x}\|_{L^2}$, you don't need a square root here which minimises

$\sqrt{\boldsymbol{x}^T \boldsymbol{L}\boldsymbol{x}}$ with respect to constant λ and $\|\boldsymbol{y} - \boldsymbol{A}\boldsymbol{x}\|_{L^2}$ (see [19, fn. 6] and [50, Fig. 2.13]). You need to fix this sentence, as L(x,lambda) is not a Lagrange multiplier – it does have a name, and you should use it and not wite sentences that say : I introduce the chicken p=7 which minimizes frog. So every solution $\boldsymbol{x}_\lambda$ is an extremum (the most regularised solution for a constant data misfit norm) and almost every sample of the posterior, which represents a feasible solution given the data, has a higher $\sqrt{\boldsymbol{x}^T \boldsymbol{L}\boldsymbol{x}}$ value and lays above the L-Curve. not accurate enough say on or above - -do any points in the posterior actually lie in the L-Cerve, I don't think so. Are you claiming that the regularised solutions are in the posterior? The L-curve could be the limit of an open set.

4.4 Comparison

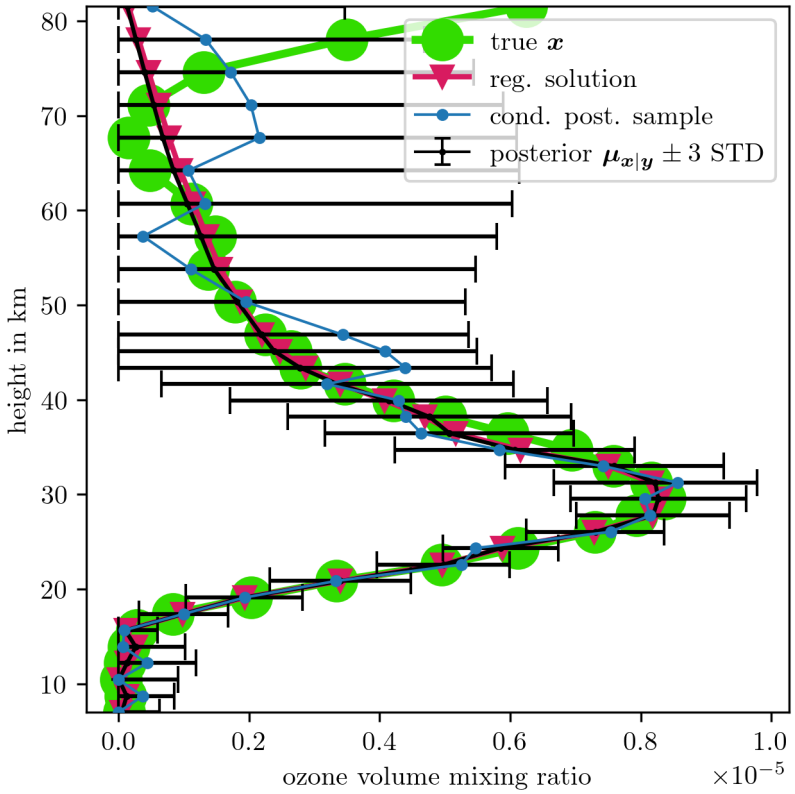


Figure 4.10: Full posterior mean and variance and one ozone sample from the full posterior. We plot the regularised solution on top of the ground truth ozone profile in green. The results are based on the approximated forward model $\boldsymbol{M}\boldsymbol{A}_L$.

Again, we calculate the full posterior mean $\boldsymbol{\mu}_{\boldsymbol{x}|\boldsymbol{y}}$, see Eq. 4.17, and covariance matrix $\boldsymbol{\Sigma}_{\boldsymbol{x}|\boldsymbol{y}}$ 4.18 as weighted expectation. We plot the results and one sample of $\pi(\boldsymbol{x}|\boldsymbol{y})$, which represents a feasible solution to this inverse problem, in Fig. 4.10, as well as the regularised solution (see next section), and one sample from the posterior. We can see that the

ground truth lies within three times of the STD (accounting for $\approx 99\%$ of all posterior samples) around the mean, except for the peak at around 80km. Compared to the previously calculated mean and variance based on the linear forward model $\boldsymbol{A}_L$ (see 4.5), the posterior distribution based on $\boldsymbol{M}\boldsymbol{A}_L$ does not differ significantly. This is expected since the difference between the linear and non-linear forward map of $\approx 1\%$ is small.

5

Affine Approximation of the Non-Linear Model

The forward map, which we introduce in Chapter 3, poses a weakly non-linear forward problem. We could tackle the non-lin earity by treating the inverse problem as a linear inverse problem and then iteratively updating the non-linear part after each parameter sample. Instead, as in Fig. **??** illustrated, we approximate the non-linear model using an affine map $\boldsymbol{M} : \boldsymbol{A}_L\boldsymbol{x} \to \boldsymbol{A}(\boldsymbol{x})$, which maps from the linear model $\boldsymbol{A}_L$ to the non-linear model $\boldsymbol{A}(\boldsymbol{x})$. It is very unclear why you are introducing this here. Then we are able to define a linear model $\boldsymbol{A} \coloneqq \boldsymbol{M}\boldsymbol{A}_L$, which approximates the non-linear model. Here, we give a brief introduction to affine maps.

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 [5, Def. 2.3.1]). no it's not In other words, an affine map does not need to map to the origin of the associated vector space. An affine map is a linear map on vector spaces, including a translation, or, in the words of my supervisor, C. F., a Taylor series of first order. it is not a linear map , the origin of the domain to the origin of the range, fix this sentence - -it makes no sense. For more information on affine spaces and maps, we refer to the books [5, 36]. don't say that, say that it is a linear map plus a constant. these are horrible books. No wonder you have a garbled idea and description. All affine maps look like y = Ax+b where A is linear and b is a constant. What's so difficult about that? I asked a LLM: Please give a simple definition of an affine map (between two vector spaces) Sure! A simple definition of an affine map between two vector spaces is: An affine map is a function between vector spaces that preserves straight lines and has the form > $f(x) = Ax + b$ > where A is a linear transformation and b is a fixed vector. So it's like a linear map, but with a shift.

Given the posterior distribution for ozone $\pi(\boldsymbol{x}|\boldsymbol{y})$, we can now approximate the non-linear forward model

$$\boldsymbol{A}(\boldsymbol{x}) \approx \boldsymbol{M}\boldsymbol{A}_L\boldsymbol{x}\,, \tag{5.1}$$

with an affine map $\boldsymbol{M}$ (see Fig. 5.1 for the summarised strategy). Here $\boldsymbol{A}(\boldsymbol{x})$ is non-linear noise-free data and $\boldsymbol{A}_L\boldsymbol{x}$ is linear noise-free data, both with ground truth pressure and temperature.

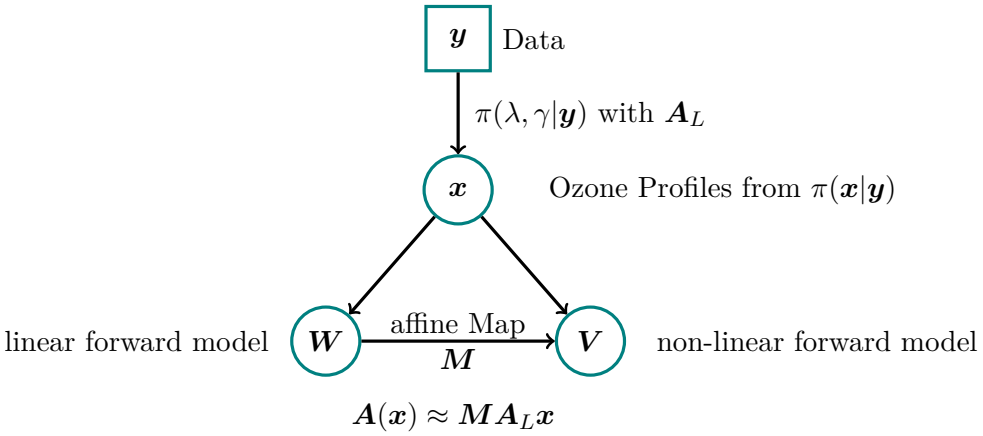


Figure 5.1: The strategy to find the affine map consists of first evaluating the marginal posterior for ozone $\pi(\lambda, \gamma|\boldsymbol{y})$ based on the linear forward model. Then we draw ozone samples from the full posterior. Based on those ozone samples, we find an affine map which approximates between noise-free data from the linear and the non-linear forward model.

Using posterior ozone samples, we generate two affine subspaces and then find the mapping between those. The subspace $\boldsymbol{W}$ is created by noise-free data based on the linear model and $\boldsymbol{V}$ by noise-free data based on the non-linear model, given m samples $\boldsymbol{x}^{(j)} \sim \pi(\boldsymbol{x}|\boldsymbol{y})$ for $j = 1, \ldots, m$. We report a relative RMS difference between $\boldsymbol{W}$ and $\boldsymbol{V}$ of about 1%, which we aim to reduce through the affine map $\boldsymbol{M}$. More specifically, the affine subspace associated with the linear forward model is

$$\boldsymbol{W} = \begin{bmatrix} | & & | & & | \\ \boldsymbol{A}_L\boldsymbol{x}^{(1)} & \cdots & \boldsymbol{A}_L\boldsymbol{x}^{(j)} & \cdots & \boldsymbol{A}_L\boldsymbol{x}^{(m)} \\ | & & | & & | \end{bmatrix} \in \mathbb{R}^{m\times m} \tag{5.2}$$

and with the non-linear forward model is

$$\boldsymbol{V} = \begin{bmatrix} | & & | & & | \\ \boldsymbol{A}(\boldsymbol{x}^{(1)}) & \cdots & \boldsymbol{A}(\boldsymbol{x}^{(j)}) & \cdots & \boldsymbol{A}(\boldsymbol{x}^{(m)}) \\ | & & | & & | \end{bmatrix} = \begin{bmatrix} — & v_1 & — \\ & \vdots & \\ — & v_j & — \\ & \vdots & \\ — & v_m & — \end{bmatrix} \in \mathbb{R}^{m \times m}\,. \tag{5.3}$$

Then we calculate the affine map

$$\boldsymbol{V}\boldsymbol{W}^{-1} = \boldsymbol{M} = \begin{bmatrix} — & r_1 & — \\ & \vdots & \\ — & r_j & — \\ & \vdots & \\ — & r_m & — \end{bmatrix} \in \mathbb{R}^{m \times m}. \tag{5.4}$$

by solving $v_j = r_j \boldsymbol{W}$ for each row r_j in $\boldsymbol{M}$, where $j = 1, \ldots, m$, using the Python function `numpy.linalg.solve`. We can do that because every measurement in the data vector $\boldsymbol{y}$ is independent of each other, and then every row v_j of $\boldsymbol{V} \in \mathbb{R}^{m \times m}$ is independent of each other as well.

We assess the affine map by calculating the relative RMS difference $\|\mathrm{vec}(\boldsymbol{M}\boldsymbol{W}) - \mathrm{vec}(\boldsymbol{V})\|_{L^2} / \|\mathrm{vec}(\boldsymbol{M}\boldsymbol{W})\|_{L^2}$ between the mapped linear noise-free data and the non-linear noise-free data, which is approximately 0.001%. In Fig. 5.2, we show the mapping for one posterior ozone sample, which has not been used to create this mapping. In other words, this is an unseen event not occuring in the training data. The relative RMS error for this approximation is roughly 0.07% and much smaller than the relative difference between noise-free linear data and non-linear data. Consequently, from here onwards, we use the approximated forward map.

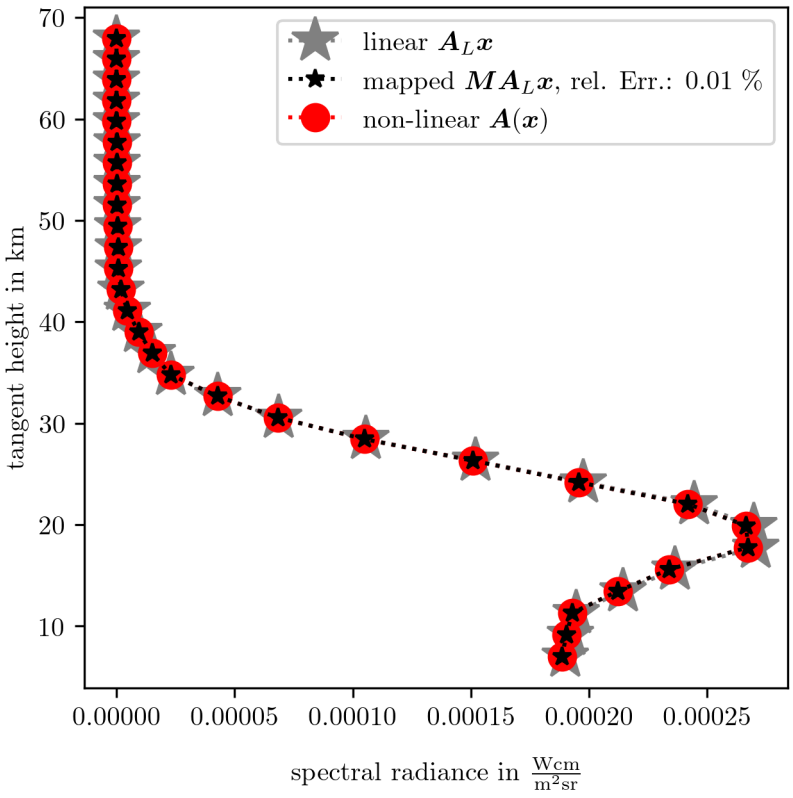


Figure 5.2: Assessment of how well we can approximate noise-free non-linear data $\boldsymbol{A}(\boldsymbol{x})$ (red circles) with noise-free linear data $\boldsymbol{A}_L\boldsymbol{x}$ (grey stars) and the previously calculated affine map $\boldsymbol{M}$. The approximated noise-free data (black stars) has a relative RMS error of $\approx 0.07\%$ compared to the true non-linear noise-free data. The ozone sample to generate this noise-free data has not been used to create the affine map.

6

Hierarchical Bayesian Model – Ozone, Pressure, Temperature

With the affine approximation, we define the linear forward model matrix

$$\boldsymbol{A} \coloneqq \boldsymbol{M}\boldsymbol{A}_L \tag{6.1}$$

using the affine map $\boldsymbol{M}$. Here, we compare the posterior distribution of ozone to a regularisation approach.

As in Sec. 4.1, we use a DAG as in Fig. 6.1 to visualise the measurement process and correlations between pressure $\boldsymbol{p}$, temperature $\boldsymbol{T}$ and ozone $\boldsymbol{x}$, which progress deterministically (dashed line) into the forward model, via $\boldsymbol{x} \times \boldsymbol{p}/\boldsymbol{T}$. Note that other variables such as absorption constant, internal partition function and the black body radiation are also dependent on temperature. Through their respective prior distributions, they generate a space of all possible noise-free data Ω, from which we observe some data, including some added normally distributed noise $\boldsymbol{\eta}$. This hierarchical Bayesian framework includes the hyper-parameters p_0, b for pressure (see Eq. 6.3), $\boldsymbol{a}, \boldsymbol{h}_T, T_0$ for temperature (see Eq. 3.11), δ for ozone smoothness and γ for noise precision. Each of those hyper-parameters is described by the hyper-prior distribution $\pi(p_0, b, T_0, \boldsymbol{h_T}, \boldsymbol{a}, \delta, \gamma)$ defined by us. Here $\boldsymbol{\theta}_\gamma, \boldsymbol{\theta}_\delta$ determine gamma distributions, e.g. $\gamma \sim \Gamma(\alpha_\gamma, \beta_\gamma)$ with $\boldsymbol{\theta}_\gamma = \{\alpha_\gamma, \beta_\gamma\}$, and $\boldsymbol{\theta}_{p_0}, \boldsymbol{\theta}_b, \boldsymbol{\theta_h}, \boldsymbol{\theta}_{T_0}, \boldsymbol{\theta_a}$ determine a normal distribution $p_0, b, \boldsymbol{h}_T, \boldsymbol{T}_0, \boldsymbol{a} \sim \pi(\boldsymbol{\theta}_{p_0}, \boldsymbol{\theta}_b, \boldsymbol{\theta_h}, \boldsymbol{\theta}_{T_0}, \boldsymbol{\theta_a})$, e.g. $b \sim \mathcal{N}(\mu_b, \sigma_b^2)$ and $\boldsymbol{\theta}_b = \{\mu_b, \sigma_b\}$. We use the approximated forward model $\boldsymbol{A}(\boldsymbol{\theta_{p,T}}) \coloneqq \boldsymbol{M}\boldsymbol{A}(\boldsymbol{\theta_{p,T}})_L$ and denote $\boldsymbol{\theta_{p,T}} \coloneqq \{p_0, b, T_0, \boldsymbol{h_T}, \boldsymbol{a}, \}$, which includes all hyper-parameter related to pressure and temperature.

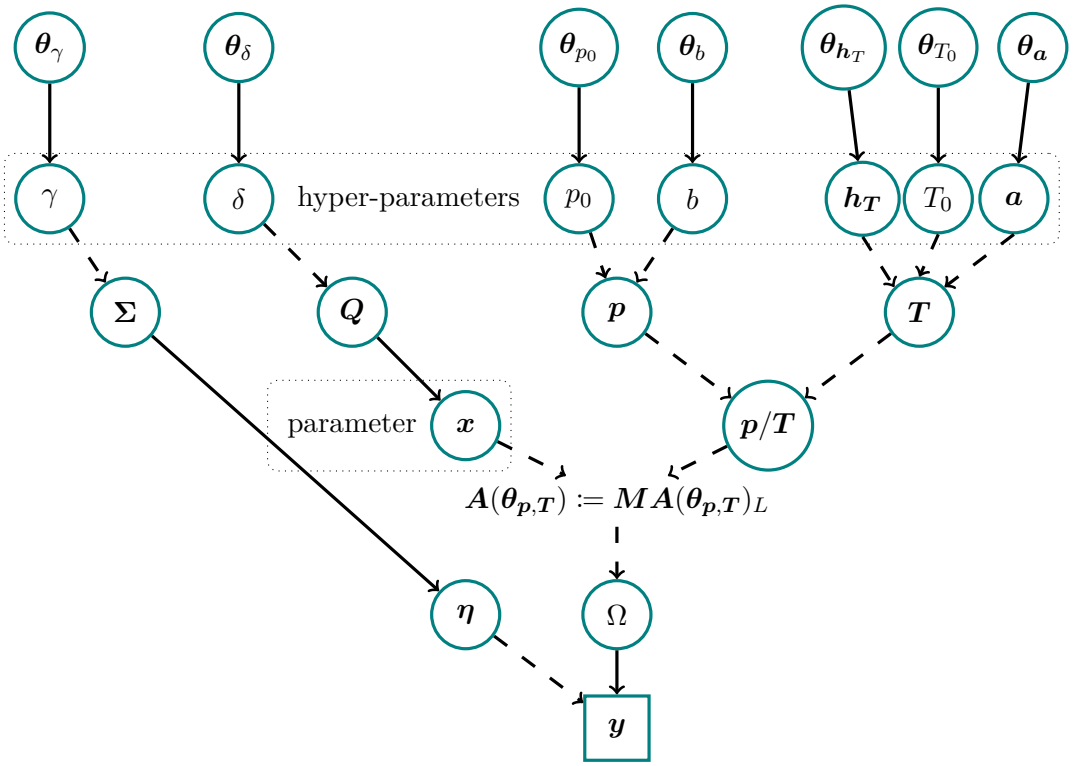


Figure 6.1: DAG of Bayesian model for ozone, pressure and temperature. The hyper-parameters $\boldsymbol{h}_T = \{h_{T,1}, h_{T,2}, h_{T,3}, h_{T,4}, h_{T,5}, h_{T,6}\}$, $\boldsymbol{a} = \{a_0, a_1, a_2, a_3, a_4, a_5, a_6\}$, T_0, b and p_0 deterministically (dotted line) describe pressure parameter $\boldsymbol{p}$ through the function in Eq. 6.3, and temperature parameter $\boldsymbol{T}$ through the function in Eq. 3.11. In this case, we choose the hyper-parameters $\pi(p_0, b, T_0, \boldsymbol{h_T}, \boldsymbol{a})$ to be a normally distributed apriori, determined by $\boldsymbol{\theta}_{\boldsymbol{h}_T}, \boldsymbol{\theta}_{\boldsymbol{a}}, \boldsymbol{\theta}_{T_0}, \boldsymbol{\theta}_b, \boldsymbol{\theta}_{p_0}$, which represent mean and variances. The ozone parameter $\boldsymbol{x}$ is statistically (solid line) described by the prior distribution $\boldsymbol{x}|\delta \sim \mathcal{N}(0, (\delta \boldsymbol{L})^{-1})$. Here, the hyper-parameter δ accounts for smoothness in the ozone profile and defines the precision matrix $\boldsymbol{Q} = \delta \boldsymbol{L}$, where $\boldsymbol{L}$ is the graph Laplacian as in Eq. 4.2. The noise covariance $\boldsymbol{\Sigma} = \gamma^{-1}\boldsymbol{I}$ of the random noise vector $\boldsymbol{\eta} \sim \mathcal{N}(0, \gamma^{-1}\boldsymbol{I})$ is defined by the noise hyper-parameter γ. As in Sec. 4.1 described, $\boldsymbol{\theta}_\delta$ and $\boldsymbol{\theta}_\gamma$ define the hyper-priors $\pi(\delta, \gamma)$. Then, we randomly observe a data set $\boldsymbol{y}$ from the space of all measurables Ω through the approximated forward model $\boldsymbol{A}(\boldsymbol{\theta}_{\boldsymbol{p},\boldsymbol{T}}) \coloneqq \boldsymbol{M}\boldsymbol{A}(\boldsymbol{\theta}_{\boldsymbol{p},\boldsymbol{T}})_L$, depending on the hyper-parameter $\boldsymbol{\theta}_{\boldsymbol{p},\boldsymbol{T}} \coloneqq \{p_0, b, \boldsymbol{h}_T, \boldsymbol{T}_0, \boldsymbol{a}\}$, including some added noise. Given the data we like to determine the posterior distribution over the hyper-parameters $\pi(p_0, b, T_0, \boldsymbol{h_T}, \boldsymbol{a}, \delta, \gamma|\boldsymbol{y})$ first and then $\pi(\boldsymbol{x}|p_0, b, T_0, \boldsymbol{h_T}, \boldsymbol{a}, \delta, \gamma, \boldsymbol{y})$, utilising the MTC scheme.

Then, we set up the hierarchical Bayesian framework

$$\boldsymbol{y}|\boldsymbol{x}, p_0, b, T_0, \boldsymbol{h_T}, \boldsymbol{a}, \delta, \gamma \sim \mathcal{N}(\boldsymbol{A}(p_0, b, T_0, \boldsymbol{h_T}, \boldsymbol{a})\,\boldsymbol{x}, \gamma^{-1}\boldsymbol{I}) \tag{6.2a}$$

$$\boldsymbol{x}|\delta \sim \mathcal{N}(\boldsymbol{0}, (\delta \boldsymbol{L})^{-1}) \tag{6.2b}$$

$$\delta \sim \Gamma(\alpha_\delta, \beta_\delta) \tag{6.2c}$$

$$\gamma \sim \Gamma(\alpha_\gamma, \beta_\gamma) \tag{6.2d}$$

$$\boldsymbol{a} \sim \mathcal{N}(\boldsymbol{\mu_a}, \boldsymbol{\Sigma_a}) \tag{6.2e}$$

$$\boldsymbol{h_T} \sim \mathcal{N}(\boldsymbol{\mu}_T, \boldsymbol{\Sigma}_{\boldsymbol{h}_T}) \tag{6.2f}$$

$$T_0 \sim \mathcal{N}(\mu_{T_0}, \sigma_{T_0}) \tag{6.2g}$$

$$p_0 \sim \mathcal{N}(\mu_{p_0}, \sigma_{p_0}) \tag{6.2h}$$

$$b \sim \mathcal{N}(\mu_b, \sigma_b) \tag{6.2i}$$

		TT bounds			
model parameters	priors	lower	upper	τ_{int}	Context
$\boldsymbol{x}$	$\mathcal{N}(0, (\delta \boldsymbol{L})^{-1})$	-	-	-	$\boldsymbol{x}$
δ	$\mathcal{T}(1, 10^{-35})$	-	-		$\boldsymbol{x}$
γ	$\mathcal{T}(1, 10^{-35})$	$2.5\,10^{14}$	$6\,10^{15}$	512 ± 30	$\boldsymbol{y}$
$\lambda = \delta/\gamma$	-	1	$2\,10^{4}$	976 ± 74	$\boldsymbol{x}$
b	$\mathcal{N}(0.174, (0.01)^2)$	0.129	0.214	864 ± 62	$\boldsymbol{p}$
$h_{T,1}$	$\mathcal{N}(11, (1.5)^2)$	5.4	16.3	270 ± 12	$\boldsymbol{T}$
T_0	$\mathcal{N}(288.15, (10)^2)$	247	326	304 ± 14	$\boldsymbol{T}$
p_0	$\mathcal{N}(1311, (20)^2)$	1237	1387	252 ± 10	$\boldsymbol{p}$
$h_{T,3}$	$\mathcal{N}(32.3, (2.5)^2)$	22.9	41.7	272 ± 12	$\boldsymbol{T}$
a_1	$\mathcal{N}(0, (0.1)^2)$	-0.38	0.38	248 ± 10	$\boldsymbol{T}$
$h_{T,2}$	$\mathcal{N}(20.1, (0.7)^2)$	17.2	22.7	268 ± 10	$\boldsymbol{T}$
a_0	$\mathcal{N}(-6.5, (0.01)^2)$	-6.54	-6.47	250 ± 10	$\boldsymbol{T}$
a_2	$\mathcal{N}(1, (0.01)^2)$	0.97	1.03	258 ± 10	$\boldsymbol{T}$
a_3	$\mathcal{N}(2.8, (0.1)^2)$	2.5	3.1	254 ± 10	$\boldsymbol{T}$
$h_{T,4}$	$\mathcal{N}(47.4, (0.5)^2)$	45.5	49.3	252 ± 10	$\boldsymbol{T}$
a_4	$\mathcal{N}(0, (0.1)^2)$	-0.38	0.38	274 ± 12	$\boldsymbol{T}$
$h_{T,5}$	$\mathcal{N}(51.4, (0.5)^2)$	49.5	53.3	274 ± 12	$\boldsymbol{T}$
a_5	$\mathcal{N}(-2.8, (0.1)^2)$	-3.18	-2.43	264 ± 10	$\boldsymbol{T}$
$h_{T,6}$	$\mathcal{N}(71.8, (3)^2)$	60.5	83.1	264 ± 10	$\boldsymbol{T}$
a_6	$\mathcal{N}(-2, (0.01)^2)$	-2.04	-1.96	262 ± 10	$\boldsymbol{T}$

Table 6.1: Summary of relevant parameter characteristics, bounds and sampling statistics. We denote $\mathcal{N}(\mu, \sigma^2)$ as the Gaussian and $\mathcal{T}(\alpha = \text{scale}, \beta = \text{rate})$ as the gamma distribution. The IACT τ_{int} is estimated as in [65] from posterior samples based on the approximated forward map.

and define a normally distributed likelihood (due to Gaussian noise) and normally distributed priors. Before we formulate the posterior distribution, we carefully define $\boldsymbol{\theta}_\gamma, \boldsymbol{\theta}_\delta, \boldsymbol{\theta}_{p_0}, \boldsymbol{\theta}_b, \boldsymbol{\theta}_{\boldsymbol{h}}, \boldsymbol{\theta}_{T_0}, \boldsymbol{\theta}_{\boldsymbol{a}}$, the hyper-prior scales, shapes, means and variances, which are explicitly given in Tab. 6.1.

6.1 Prior Modelling

We start by describing the pressure $\boldsymbol{p}$ in between $h_{L,0} \approx 7$km and $h_{L,n} \approx 82$km with an exponential function

$$p(h) = \exp(-b\,h)\; p_0 \quad , h_{L,0} \leq h \leq h_{L,n} \tag{6.3}$$

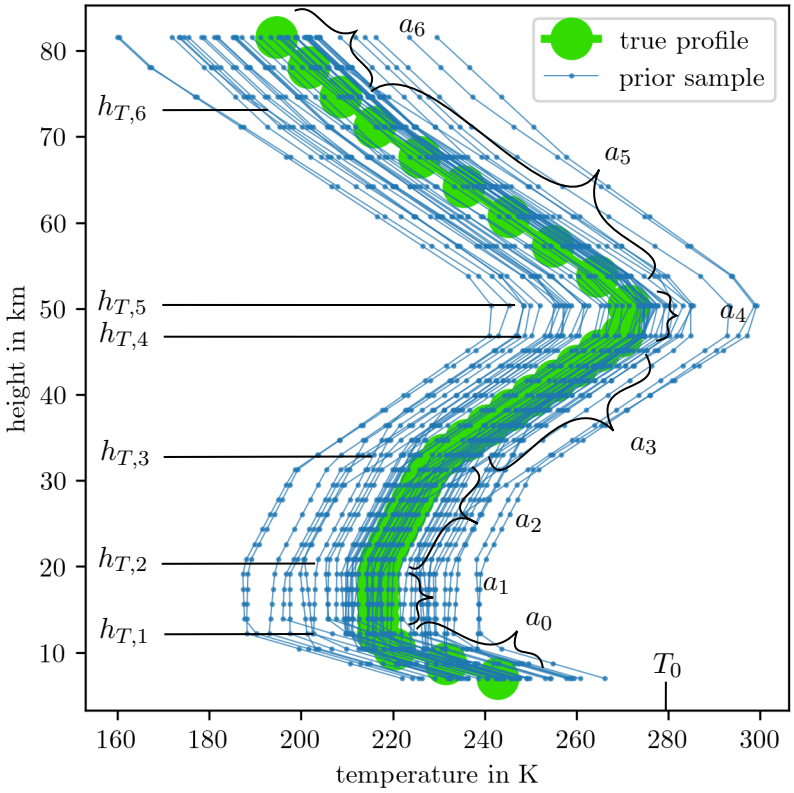


Figure 6.2: Prior samples from the hyper-prior distribution of $\boldsymbol{h}_T$, $\boldsymbol{a}$ and T_0, as defined in Tab. 6.1, where we calculate $\boldsymbol{T}$ according to the function in Eq. 3.11.

depending on two hyper-parameters p_0, b, see Fig. 6.3. Similarly, the temperature as described in Eq. 3.11 can be parametrised with 14 hyper-parameters $\boldsymbol{h}_T = \{h_{T,1}, h_{T,2}, h_{T,3}, h_{T,4}, h_{T,5}, h_{T,6}\}$, $\boldsymbol{a} = \{a_0, a_1, a_2, a_3, a_4, a_5, a_6\}$ and T_0 (see Fig. 6.2 and Eq. 3.11). To complete the model, we have to define sensible hyper-prior variances and means of the normally distributed hyper-prior distribution for pressure and temperature-related hyper-parameters, where $\pi(\delta, \gamma)$ are the same gamma distributions as previously defined in Sec. 4.1.1. We tune the normal distribution $\pi(\boldsymbol{h}_T)$, so that the temperature profile maintains its structure, $h_{T,i} < h_{T,i+1}$ for $i = 1, \ldots, 5$ (see Fig. B.7) and set $\pi(\boldsymbol{a})$ to a normal distribution as well. Similarly, we set $\pi(T_0)$ to a normal distribution, so that it mimics a daily temperature variability of roughly 30K. We choose those rather informative hyper-prior distributions, because we find (see Fig. 6.4) that the data is uninformative about the temperature profile. The hyper-prior distribution $\pi(p_0, b)$ for pressure-related hyper-parameters is also normally distributed, but with rather large variance σ_b^2, where p_0 has a variability of around 80hPa, close to what we can observe when looking at weather data. We set means of the normal distribution $\pi(p_0, b, T_0, \boldsymbol{h_T}, \boldsymbol{a})$ to the ground truth values of $\boldsymbol{T}$ and $\boldsymbol{p}$ which for $\boldsymbol{p}$ we find with the python function `scipy.optimize.curve_fit` (see Tab. 6.1).

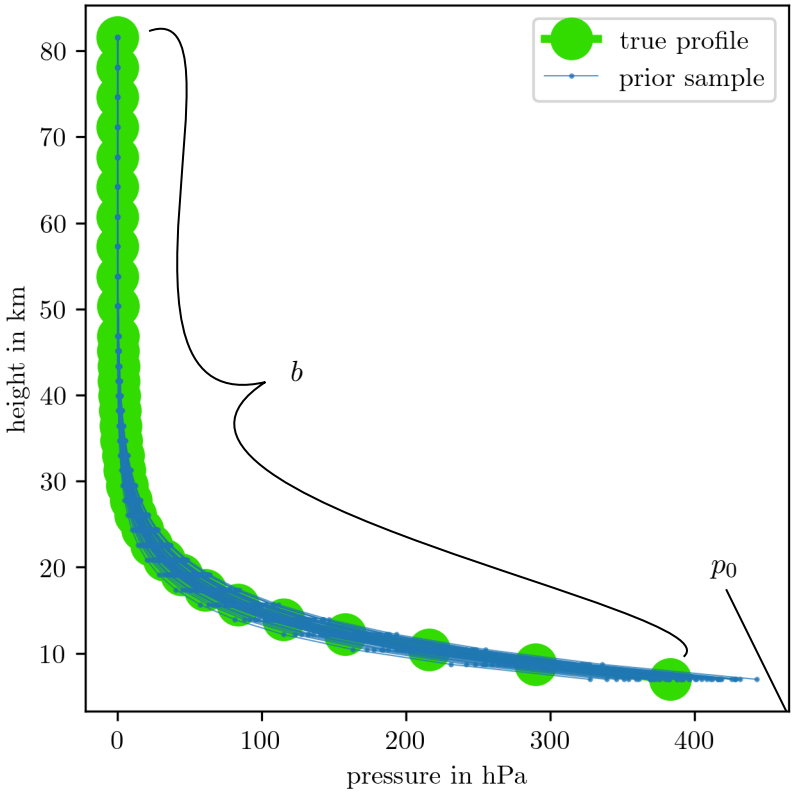


Figure 6.3: Prior samples from the hyper-prior distribution of b and p_0 as defined in Tab. 6.1, where we calculate $\boldsymbol{p}$ according to the function in Eq. 6.3.

We plot prior samples of the pressure $\boldsymbol{p}$ in Fig. 6.3, the temperature $\boldsymbol{T}$ in Fig. 6.2 and the ratio $\boldsymbol{p}/\boldsymbol{T}$ in Fig. 6.4 against the ground truth profiles. Additionally, we plot prior samples of $1/\boldsymbol{T}$ in Fig. B.8. Here we already observe that $\boldsymbol{p}/\boldsymbol{T}$ inherits the structure of the pressure function and hence the model is uninformative about the temperature.

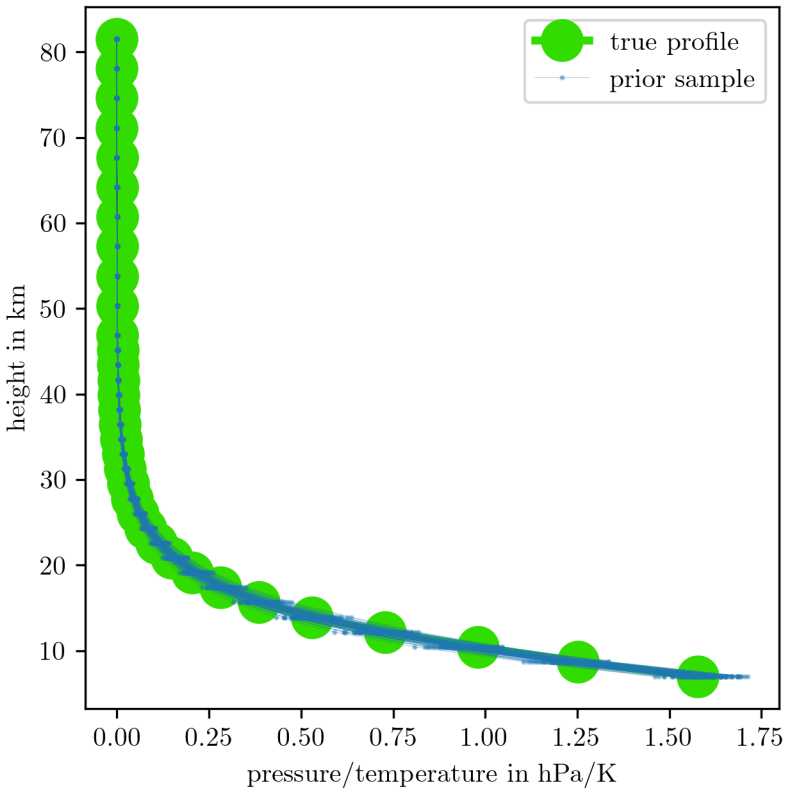


Figure 6.4: Prior samples from the hyper-prior distribution of $\boldsymbol{h}_T$, $\boldsymbol{a}$ and T_0 for temperature as in Eq. 3.11 and b and p_0 for pressure as in Eq. 6.3. We plot $\boldsymbol{p}/\boldsymbol{T}$. The hyper-priors are defined in Tab. 6.1.

6.2 Marginal Posterior distribution

Here, we define the marginal and then the full conditional posterior distribution for the described Bayesian model. We either use the t-walk algorithm [7] to draw samples from $\pi(p_0, b, T_0, \boldsymbol{h_T}, \boldsymbol{a}, \lambda, \gamma|\boldsymbol{y})$ or we utilise a TT approximation on a predefined grid to generate samples via the SIRT method with an MH correction step. In doing so, we guide the reader through the procedure and point out some key aspects of how we obtain an efficient TT approximation. Lastly, we use the RTO method to draw samples from the full conditional posterior $\pi(\boldsymbol{x}|p_0, b, T_0, \boldsymbol{h_T}, \boldsymbol{a}, \lambda, \gamma, \boldsymbol{y})$.

The marginal posterior is given as

$$\pi(\boldsymbol{\theta_{p,T}}, \lambda, \gamma|\boldsymbol{y}) \propto \lambda^{n/2}\gamma^{m/2} \exp\left\{-\frac{1}{2}g(\boldsymbol{\theta_{p,T}}, \lambda) - \frac{\gamma}{2}f(\boldsymbol{\theta_{p,T}}, \lambda)\right\}\pi(\boldsymbol{\theta_{p,T}}, \lambda, \gamma)\,, \tag{6.4}$$

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

$$f(\boldsymbol{\theta_{p,T}}, \lambda) = \boldsymbol{y}^T\boldsymbol{y} - (\boldsymbol{A}(\boldsymbol{\theta_{p,T}})^T\boldsymbol{y})^T(\boldsymbol{A}(\boldsymbol{\theta_{p,T}})^T\boldsymbol{A}(\boldsymbol{\theta_{p,T}}) + \lambda\boldsymbol{L})^{-1}(\boldsymbol{A}(\boldsymbol{\theta_{p,T}})^T\boldsymbol{y})\,, \tag{6.5a}$$

$$\text{and } g(\boldsymbol{\theta_{p,T}}, \lambda) = \log\det\left(\boldsymbol{A}(\boldsymbol{\theta_{p,T}})^T\boldsymbol{A}(\boldsymbol{\theta_{p,T}}) + \lambda\boldsymbol{L}\right). \tag{6.5b}$$

For each evaluation of $\pi(\boldsymbol{\theta_{p,T}}, \lambda, \gamma|\boldsymbol{y})$ we compose $\boldsymbol{A}_L(\boldsymbol{\theta_{p,T}})$ as in Chapter 3, and calculate f and g directly using the Cholesky decomposition.

6.2.1 Sampling from marginal posterior

T-walk Sampler as a Black Box If the number of hyper-parameters is large, we sample from the marginal posterior $\pi(\boldsymbol{\theta}|\boldsymbol{y})$ using the t-walk sampler as by Christen and Fox [7], because it is easy-to-use and a quick-to-implement sampler. The t-walk chooses between four different types of steps on the target distribution and is employed as a black-box algorithm in default settings, requiring the specification of the number of samples, burn-in period, support region, and the target distribution. Convergence to the target distribution is guaranteed by the construction of this algorithm (see [7]). Oh, are you defining some kind of decision tree about what kind of sampler you run. Did you introduce that ? At the moment this reads like a detective novel – I have no idea what is going on - -I have to guess. Not a good thing to have the reader guessing.

For a ground truth, we run the t-walk [7] algorithm on $\pi(\boldsymbol{\theta_{p,T}}, \lambda, \gamma|\boldsymbol{y})$, where we set our objective to generate 1000 independent samples from the marginal posterior. We bound the maximum IACT (see Tab. 6.1 and Fig. B.10 to Fig. B.27) by 1100, so we run the t-walk for $N = 1000 \times 1100$ steps with a burn-in period of $N_{\text{burn-in}} = 100 \times 1100$. We initialise the Python implementation of the t-walk [8] around the hyper-prior mean values and the mode of $\pi(\lambda, \gamma|\boldsymbol{y})$. We take a total number of $N' = N + N_{\text{burn-in}} = 1210000$ steps in ≈ 10 mins within bounds given by the iteratively defined TT grid (see Tab. 6.1). We plot the resulting histograms in Fig. 6.7 to Fig. 6.11 and the trace of the samples in Fig. B.9.

6.2.2 TT Approximation of marginal posterior

The aim now is to approximate the square root of the marginal posterior

$$\sqrt{\pi(\lambda, \boldsymbol{\theta}_{\boldsymbol{p},\boldsymbol{T}}, \gamma|\boldsymbol{y})} \propto \exp\Big\{0.5 \log \pi(\lambda, \boldsymbol{\theta}_{\boldsymbol{p},\boldsymbol{T}}, \gamma|\boldsymbol{y}) + c\Big\}, \tag{6.6}$$

where we introduce a "normalisation constant" $c = -50$ to avoid under or overflow and stay within computer precision. In doing so we run the `tt.cross.rectcross.rect_cross.cross` function from the `ttpy` python package [39]. We set the grid according to the results of the t-walk. If we compute the marginal as in Sec. 2.2, we set $\xi = 1/\boldsymbol{\lambda}(\mathcal{X})$ and $\boldsymbol{\lambda}(x) = 1$ so that for Cartesian basis $\boldsymbol{M}_k = \mathrm{diag}(\boldsymbol{\lambda}_k(\mathcal{X}_k))$. To draw samples from this TT approximation we use the SIRT-MH scheme as in Sec. 2.2.2.

Correlation structure

First, we order the hyper-parameters according to their correlation structure to improve the efficiency of the TT approximation. Specifically, we arrange the hyper-parameter space $\mathcal{X}_\gamma \times \mathcal{X}_\lambda \times \mathcal{X}_b \times \cdots$ in such a way that highly correlated hyper-parameter pairs are adjacent and directly linked through their shared TT rank. In Fig. 6.5 we plot 1000 independent samples drawn via the SIRT-MH scheme from the TT approximation of $\sqrt{\pi(\boldsymbol{\theta}_{\boldsymbol{p},\boldsymbol{T}}, \lambda, \gamma|\boldsymbol{y})}$ and the Pearson correlation coefficient between hyper-parameter pairs. A coefficient close to 1 or -1 indicates strong correlation, while values near zero suggest weak or no correlation. We observe that the hyper-parameters λ and b, and λ and γ are highly correlated. Additionally, $h_{T,1}$ and T_0 describing the temperature at low altitudes (strong signal) are mildly correlated to b as well. This is because $h_{T,1}$ and T_0 influence "the smoothness" of $\boldsymbol{p}/\boldsymbol{T}$, which is hard to see in Fig. 6.4. Interestingly, p_0 appears largely uncorrelated with other hyper-parameters, while b is the key parameter linking pressure to ozone and temperature. Hyper-parameters describing temperature at higher altitudes are very much uncorrelated and the IACTs in Tab. 6.1 agree with those results.

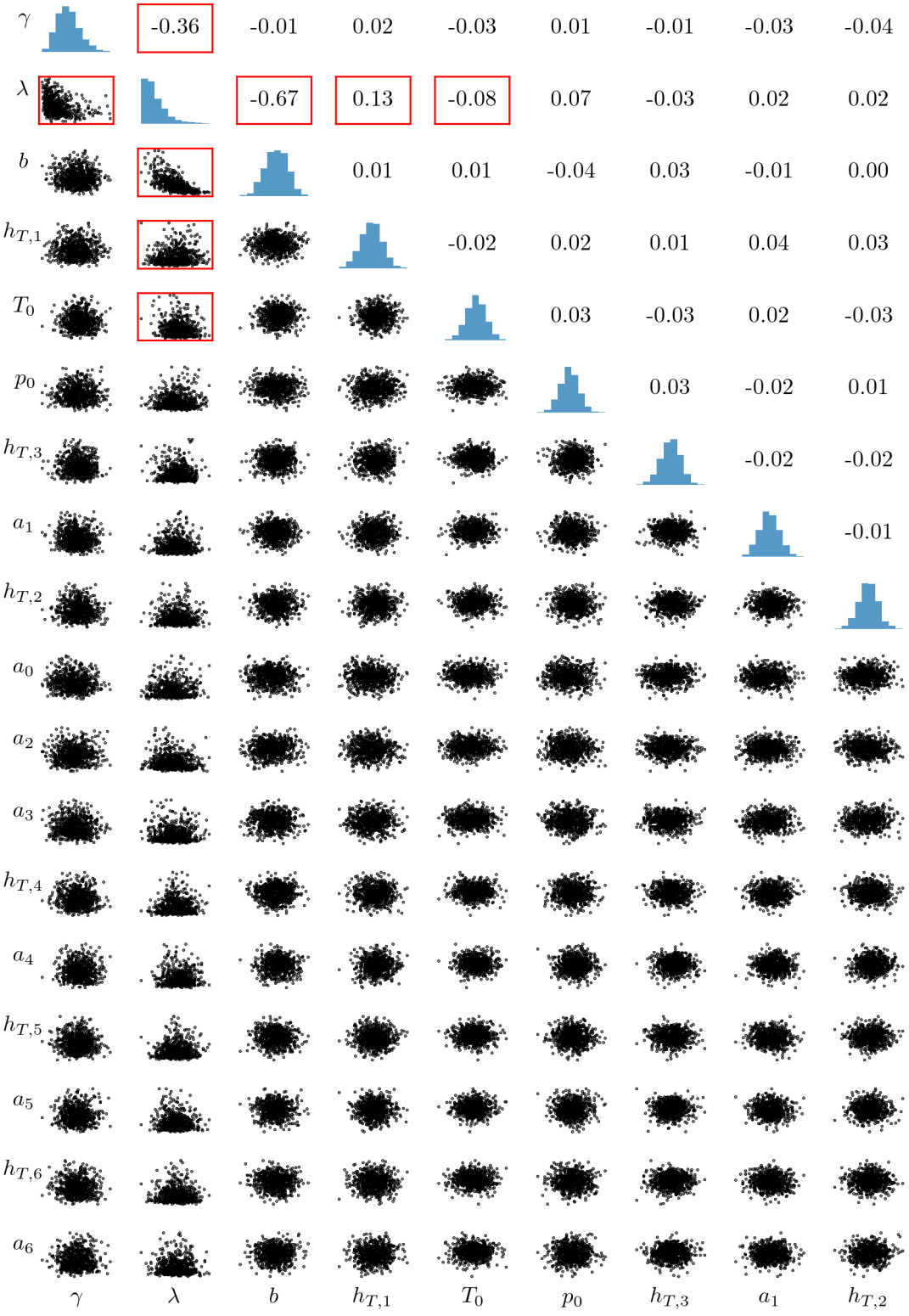
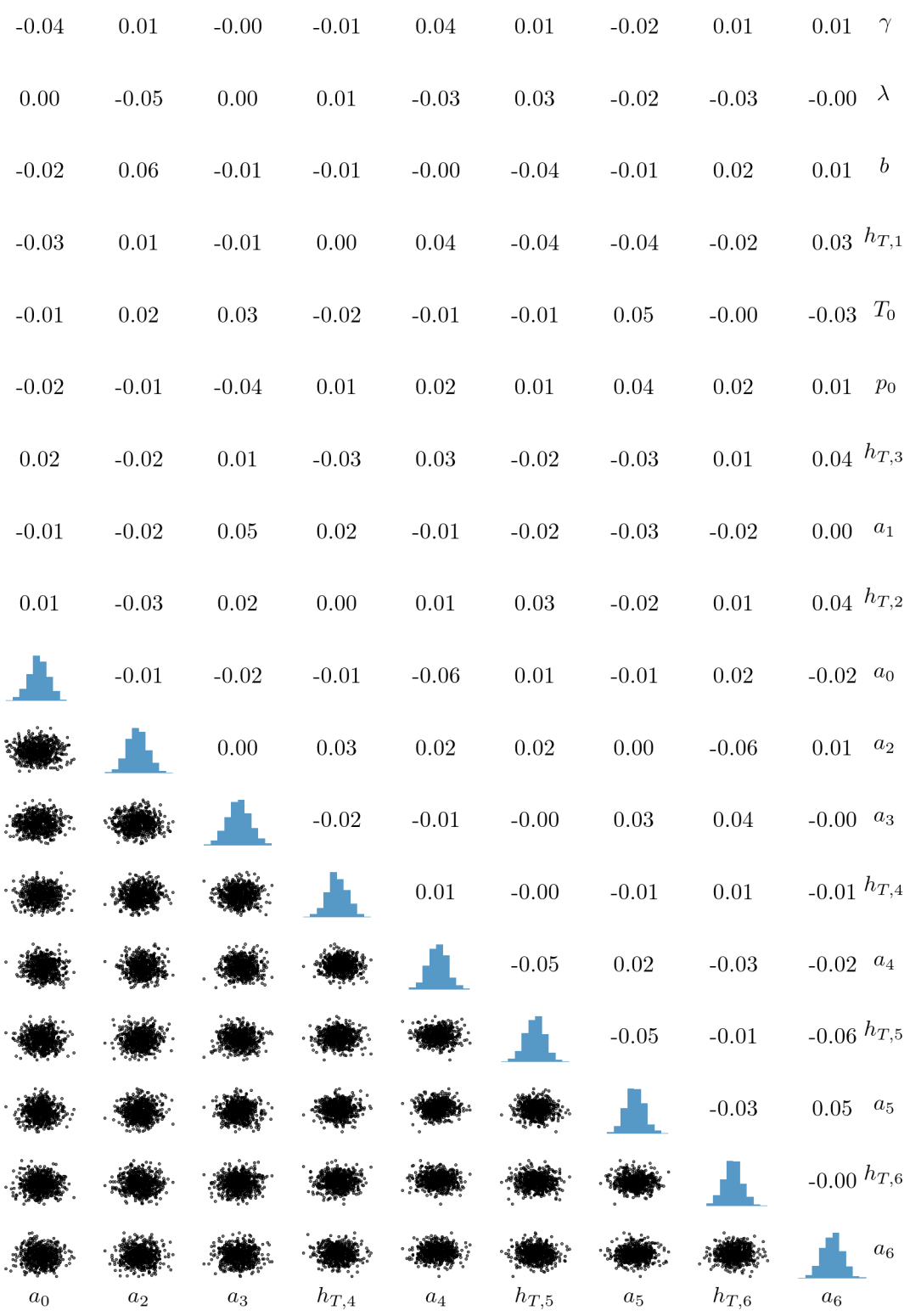


Figure 6.5: Plot of 1000 independent samples from TT approximation of $\sqrt{\pi(\boldsymbol{\theta}_{\boldsymbol{p},\boldsymbol{T}}, \lambda, \gamma|\boldsymbol{y})}$ via SIRT-MH scheme. We plot the Pearson correlation coefficient ranging from -1 to 1 for each hyper-parameter pair.



Correlation plot of samples from TT-approximation of $\sqrt{\pi(p_0, b, T_0, \boldsymbol{h_T}, \boldsymbol{a_T}|\boldsymbol{y}, \gamma, \boldsymbol{x})}$ via SIRT scheme.

Find optimal rank and grid size

Next we aim to determine the optimal rank and grid size to accurately approximate the marginal posterior, while decreasing the number of function evaluation. We set the number of grid points to $n = 150$ and calculate different error measures for deceasing number of ranks to find the optimal number of ranks, where we compare to true marginal posterior function values and 1000 independent t-walk samples from that distribution. Then we fix a small but tolerable rank and decrease the number of grid points until sufficient accuracy.

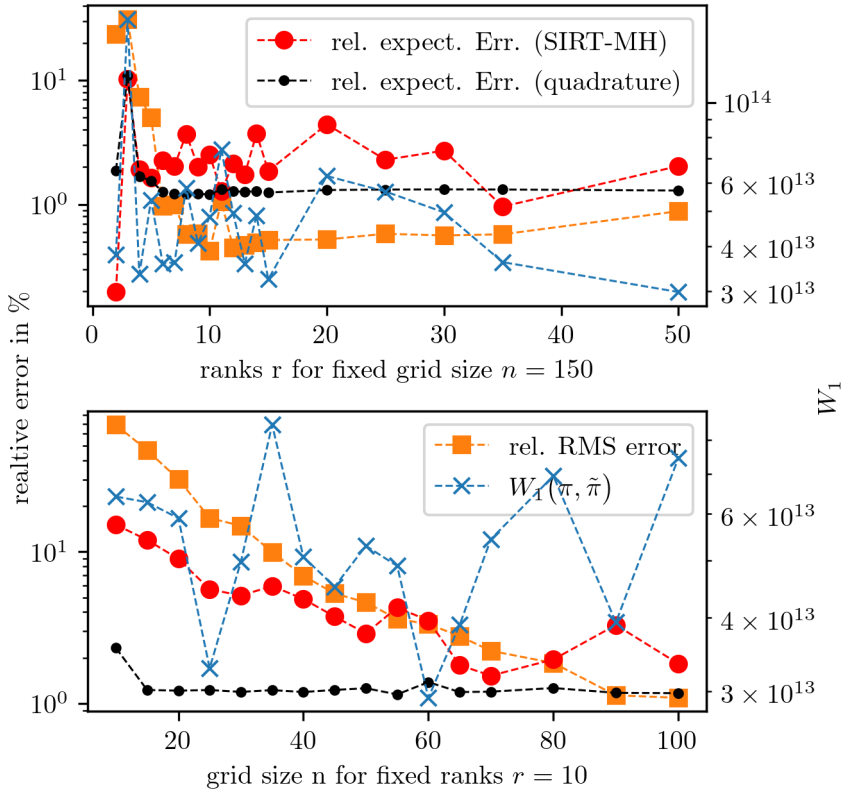


Figure 6.6: Given a TT approximation of $\sqrt{\pi(\lambda, \boldsymbol{\theta}_{\boldsymbol{p},\boldsymbol{T}}, \gamma|\boldsymbol{y})}$, we calculate the relative RMS error (orange squares) and the 1-Wasserstein distance (blue cross) between approximated values at sample points provided by the SIRT-MH and the true function values. We calculate the relative RMS error between the sample mean provided by the t-walk and mean values for the hyper-parameters calculated by quadrature (black dots), where we use the marginal function from the TT approximation as weights. Additionally, we plot the relative RMS error between the sample-based mean from the SIRT-MH and the t-walk (red circles).

For stable and comparable results, we do five sweeps in the `tt.cross.rectcross.rect_cross.cross` python function initialised with a random TT. Then we draw 1000 independent samples from the TT approximation of the marginal posterior via the SIRT-MH scheme. To calculate the 1-Wasserstein distance, as in Eq. 2.48, between SIRT-MH samples weighted with the TT approximation of marginal posterior and the t-walk samples, weighted by the true marginal posterior values, we use

the `SamplesLoss("sinkhorn", p=1, blur=0.05, scaling=0.8)` function with default settings from the Python package `geomloss` [17]. This provides the unbiased Sinkhorn divergence, which converges towards the Wasserstein distance and can be understood as the generalised Quicksort algorithm [16]. Here, p = 1 defines the distance measure $\|\boldsymbol{x} - \tilde{\boldsymbol{x}}\|_{L^2}$, the blur parameter is an entropic penalty and the scaling parameter specifies the trade-off between speed (scaling < 0.4) and accuracy (scaling > 0.9) [17]. Additionally, we use the marginal functions of each TT approximation to calculate the means $\boldsymbol{\mu}_{\text{TT}} \in \mathbb{R}^{18}$ of each hyper-parameter by weighted expectations. Then we obtain the relative RMS difference $\|\boldsymbol{\mu}_{\text{TT}} - \boldsymbol{\mu}_{\text{t-walk}}\|_{L^2} / \|\boldsymbol{\mu}_{\text{t-walk}}\|_{L^2}$. Here $\boldsymbol{\mu}_{\text{t-walk}}$ denotes the "true sample-based means" from the t-walk. Further, we calculate the relative RMS between the SIRT-MH sample-based mean $\boldsymbol{\mu}_{\text{SIRT-MH}}$ and $\boldsymbol{\mu}_{\text{t-walk}}$, and the relative RMS error at SIRT-MH samples compared to ground truth function values (Eq. 2.44).

We plot all of these measures in Fig. 6.6, where we observe that a rank $r = 10$ is sufficient because the error measures are relatively stable for $r \geq 10$. Then we decrease the grid size and decide that for $n \geq 20$ the relative differences of sample-based means to $\boldsymbol{\mu}_{\text{t-walk}}$ (red circles in Fig. 6.6) and the RMS error at the SIRT-MH samples (orange squares in Fig. 6.6) around 20% is good enough. We relate a more accurate interpolation in between grid points to decreasing sample-based relative errors for an increasing number of grid points, since the chosen linear interpolation (see Eq. 2.41) is a rather rudimentary choice. The quadrature-based relative expectation error (black dots in Fig. 6.6) is almost constant for ranks ≥ 7 and grid sizes ≥ 15. Since the hyper-parameters have different length scales, we are only interested in the trend of the 1-Wasserstein distance (blue crosses in Fig. 6.6). The 1-Wasserstein distance is quite fluctuant but decreases with increasing ranks and stays within a similar range for decreasing grid size.

Further, we decrease the number of functions evaluations and define ranks $r = [1, 10, 10, 10, 10, 10, 5, 5, 5, 5, 5, 3, 2, 2, 2, 2, 2, 2, 1]$ harvesting the correlation structure of $\pi(\boldsymbol{\theta}_{\boldsymbol{p},\boldsymbol{T}}, \lambda, \gamma | \boldsymbol{y})$ even more. We do one sweep in the `tt.cross.rectcross.rect_cross.cross`, reducing the computation time to $\approx$ 7s and the number of function evaluations to 24120, where we initialise at a previously calculated approximation. We report an average IACT (provided by [64, 28]) of $\approx 1.2 \pm 0.2$ for the samples drawn via the SIRT-MH scheme, which means that we need two function evaluations per independent sample, once the TT approximation is available. To draw 1000 independent samples, including generating a TT approximation, takes $\approx$ 30s, and we report a relative RMS error of $\approx$ 25% evaluated over those 1000 independent samples. Additionally, we report a relative RMS error of $\leq$ 1% on 1000 randomly chosen grid points, indicating that the linear interpolation causes most of the approximation error. We plot the marginals for each hyper-parameter in Fig. 6.7 to Fig. 6.12 and samples in

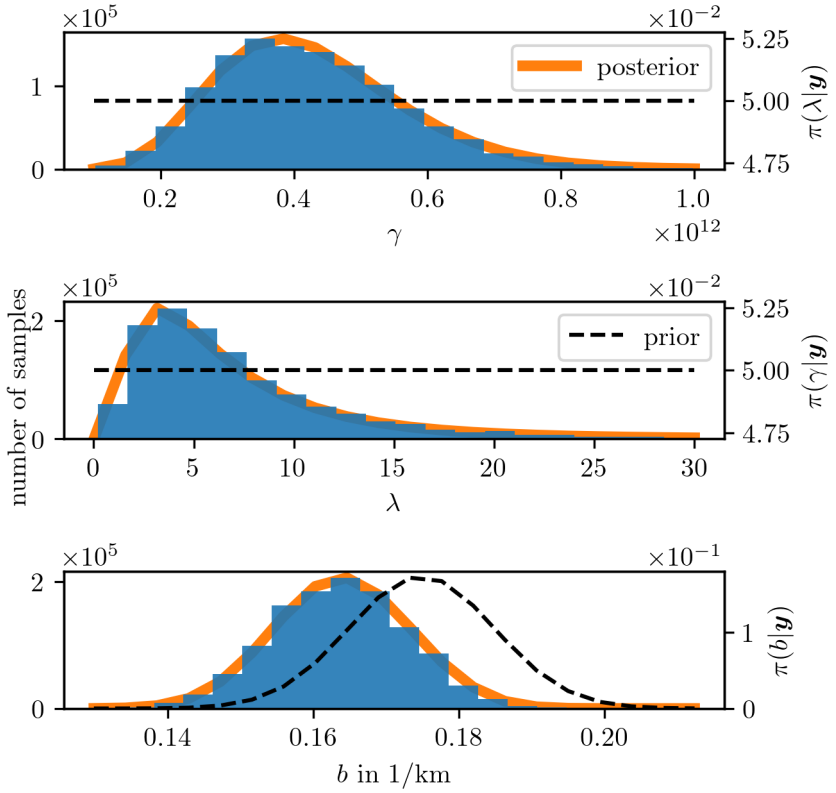


Figure 6.7: TT approximation of the marginal posterior in orange and the samples as a histogram as well as the prior distribution with a dotted line.

Fig. 6.5. We observe that, besides λ and γ, only the marginal posterior of the b hyper-parameter is seriously affected by the data and has significantly changed compared to the hyper-prior distribution.

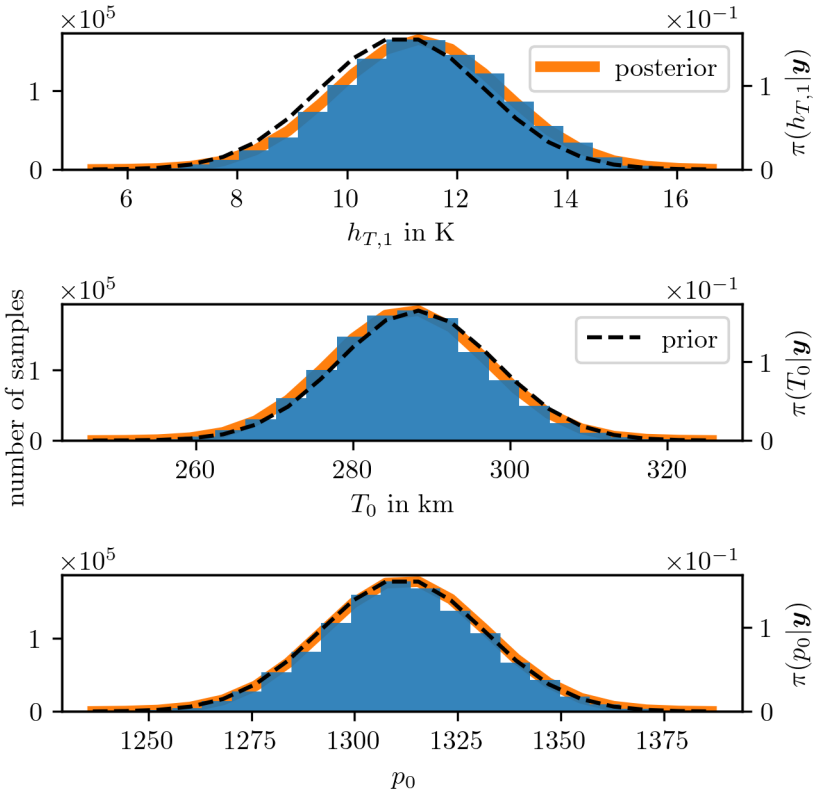


Figure 6.8: TT approximation of the marginal posterior in orange and the samples as a histogram as well as the prior distribution with a dotted line.

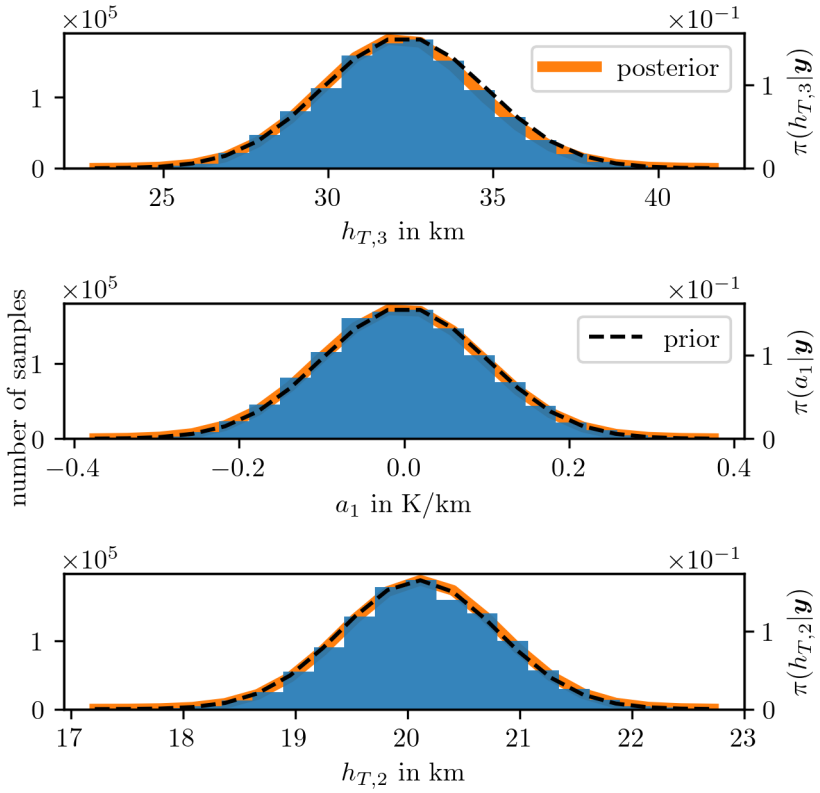


Figure 6.9: TT approximation of the marginal posterior in orange and the samples as a histogram as well as the prior distribution with a dotted line.

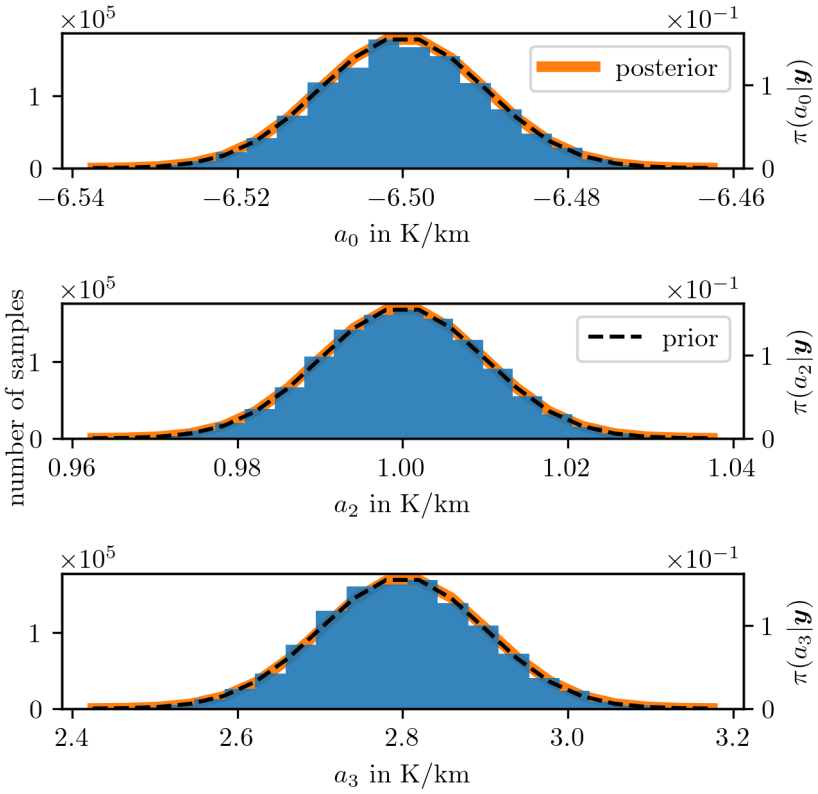


Figure 6.10: TT approximation of the marginal posterior in orange and the samples as a histogram as well as the prior distribution with a dotted line.

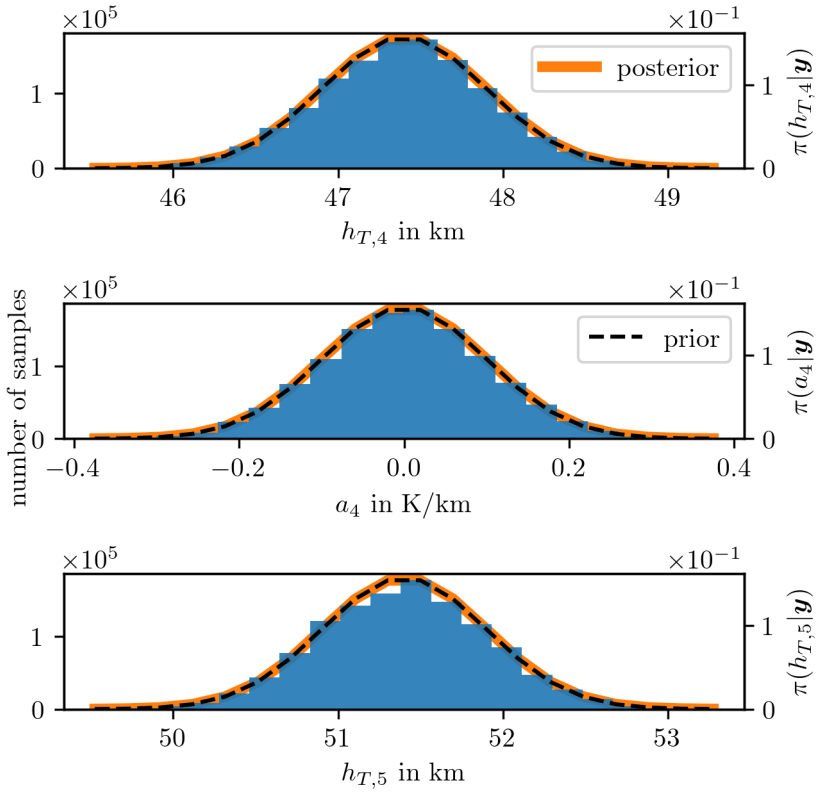


Figure 6.11: TT approximation of the marginal posterior in orange and the samples as a histogram as well as the prior distribution with a dotted line.

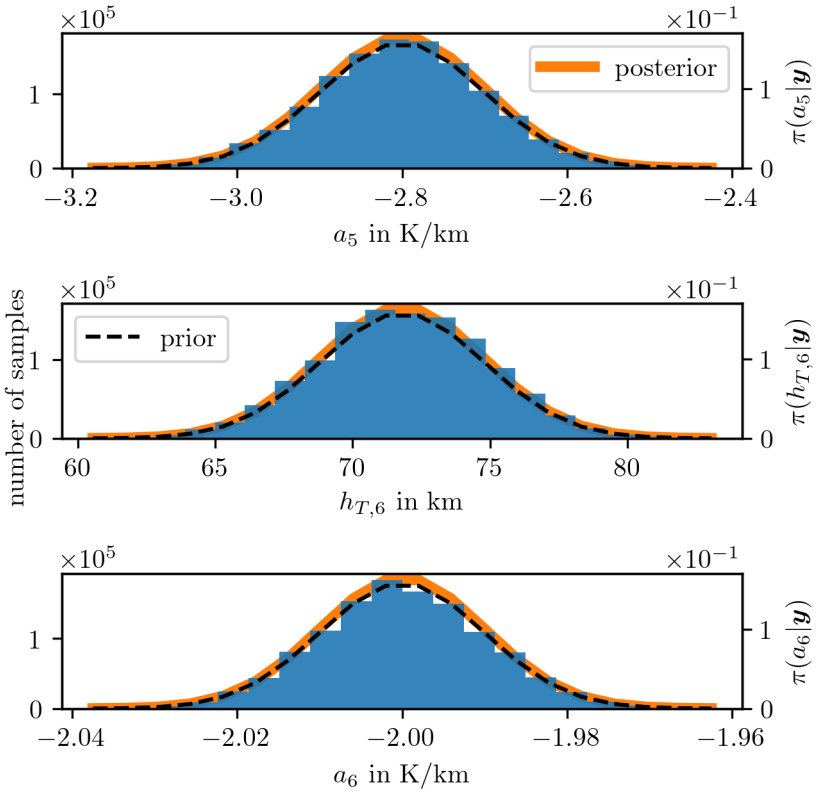


Figure 6.12: TT approximation of the marginal posterior in orange and the samples as a histogram as well as the prior distribution with a dotted line.

6.2.3 Posterior Pressure and Temperature

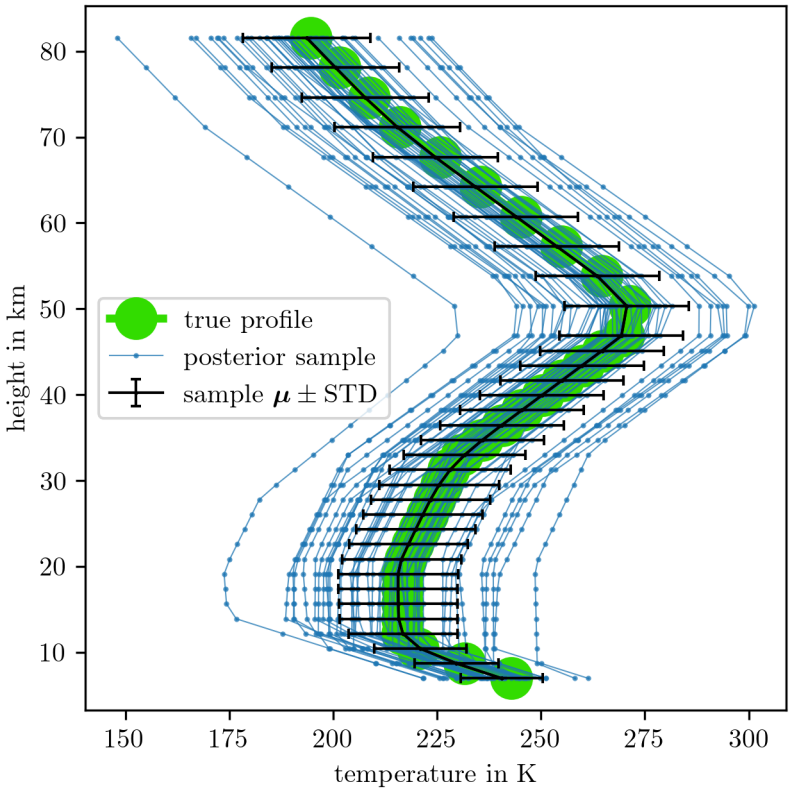


Figure 6.13: According to the hyper-parameter samples from the marginal posterior distribution (see Fig. 6.8 to Fig. 6.12) we plot the corresponding posterior temperature profile as given by Eq. 3.11.

6.3 Full conditional posterior distribution

We use the RTO method (see Sec. 6.3.1) to obtain ozone samples from the full conditional posterior and sample hyper-parameter samples directly from the marginal posterior to compute posterior temperature and pressure profiles according to their respective function (see Eq. 3.11 and Eq. 6.3). We plot posterior profiles of ozone in Fig. 6.15, temperature in Fig. 6.13 and pressure in Fig. 6.14.

6.3.1 Randomise then Optimise

If it is computationally not feasible to calculate the mean and the covariance matrix of the full posterior (see Eq. 2.17 and 2.18) via quadrature due to a large number of hyper-parameters, e.g. $\boldsymbol{\theta} \in \mathbb{R}^{n_{\boldsymbol{\theta}}}$ and $n_{\boldsymbol{\theta}} \geq 4$, we need an alternative way to draw samples from $\pi(\boldsymbol{x}|\boldsymbol{\theta}, \boldsymbol{y})$. For the linear-Gaussian Bayesian model, we condition on $\boldsymbol{\theta}$ and draw samples from the full conditional normal posterior distribution $\pi(\boldsymbol{x}|\boldsymbol{\theta}, \boldsymbol{y})$ using the RTO method [2].

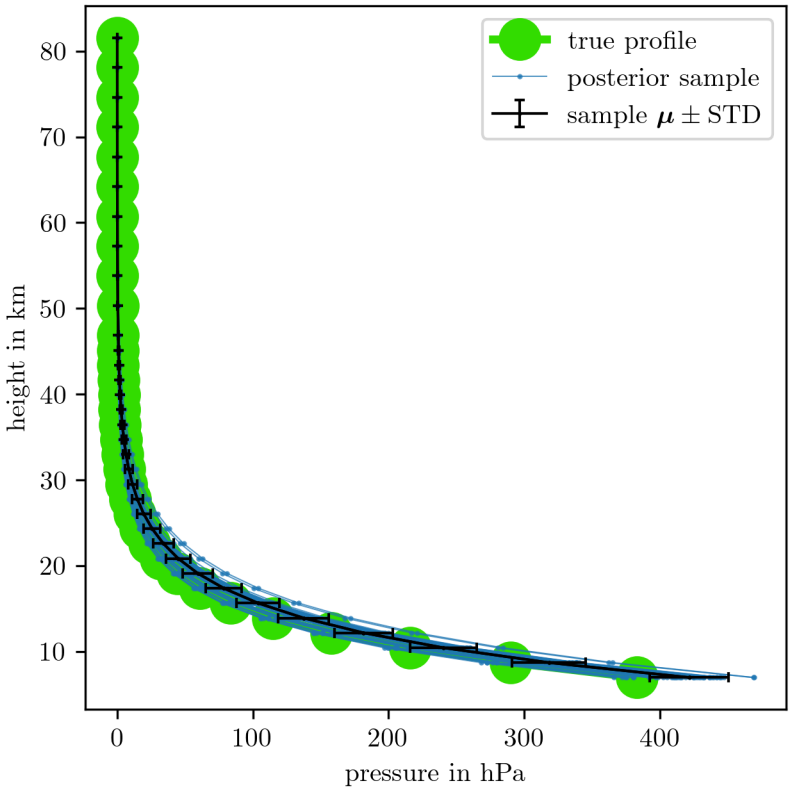


Figure 6.14: According to the hyper-parameter samples from the marginal posterior distribution (see Fig. 6.7 and Fig. 6.8) we plot the corresponding posterior pressure profile as given by Eq. 6.3.

OK, another condition and another method. You definitelu need some kind of explanatory introduction to this, So these are going to be specific elgorithms, but this section is 'general', so these end up being waffly and of no clear purpose. You have some work to do.

The full conditional posterior, as in Eq. 2.15, can be rewritten as

$$\pi(\boldsymbol{x}|\boldsymbol{\theta},\boldsymbol{y}) \propto \pi(\boldsymbol{y}|\boldsymbol{x},\boldsymbol{\theta})\pi(\boldsymbol{x}|\boldsymbol{\theta}) \tag{6.7}$$

$$\propto \exp\left(-\frac{1}{2}(\boldsymbol{A_\theta}\,\boldsymbol{x}-\boldsymbol{y})^T\boldsymbol{\Sigma}_{\boldsymbol{\theta}}^{-1}(\boldsymbol{A_\theta}\,\boldsymbol{x}-\boldsymbol{y})\right)\exp\left(-\frac{1}{2}(\boldsymbol{\mu}-\boldsymbol{x})^T\boldsymbol{Q_\theta}(\boldsymbol{\mu}-\boldsymbol{x})\right), \tag{6.8}$$

$$= \exp\left(-\frac{1}{2}\left\|\hat{\boldsymbol{A}}\boldsymbol{x}-\hat{\boldsymbol{y}}\right\|_{L^2}^2\right), \tag{6.9}$$

where we define

$$\hat{\boldsymbol{A}} \coloneqq \begin{bmatrix}\boldsymbol{\Sigma}_{\boldsymbol{\theta}}^{-1/2}\boldsymbol{A_\theta}\\ \boldsymbol{Q}_{\boldsymbol{\theta}}^{1/2}\end{bmatrix}, \quad \hat{\boldsymbol{y}} \coloneqq \begin{bmatrix}\boldsymbol{\Sigma}_{\boldsymbol{\theta}}^{-1/2}\boldsymbol{y}\\ \boldsymbol{Q}_{\boldsymbol{\theta}}^{1/2}\boldsymbol{\mu}\end{bmatrix} \quad [4,\,3]\,, \tag{6.10}$$

with $\boldsymbol{A}(\boldsymbol{\theta}) \coloneqq \boldsymbol{A_\theta}$, $\boldsymbol{Q}(\boldsymbol{\theta}) \coloneqq \boldsymbol{Q_\theta}$ and $\boldsymbol{\Sigma}^{-1}(\boldsymbol{\theta}) \coloneqq \boldsymbol{\Sigma}_{\boldsymbol{\theta}}^{-1}$, which are all dependent on the hyper-parameters $\boldsymbol{\theta}$. A sample $\boldsymbol{x}^{(k)} \sim \pi(\boldsymbol{x}|\boldsymbol{\theta},\boldsymbol{y})$ from the full conditional posterior is

obtained by minimising the following equation with respect to $\boldsymbol{x}$:

$$\boldsymbol{x}^{(k)} = \arg\min_{\boldsymbol{x}} \|\hat{\boldsymbol{A}}\boldsymbol{x} - (\hat{\boldsymbol{y}} + \boldsymbol{b})\|_{L^2}^2, \quad \boldsymbol{b} \sim \mathcal{N}(\boldsymbol{0}, \mathbf{I}), \tag{6.11}$$

where we add a random perturbation $\boldsymbol{b}$. Similar to Section **??**, this expression becomes

$$\left(\boldsymbol{A}_{\boldsymbol{\theta}}^T \boldsymbol{\Sigma}_{\boldsymbol{\theta}}^{-1} \boldsymbol{A}_{\boldsymbol{\theta}} + \boldsymbol{Q}_{\boldsymbol{\theta}}\right) \boldsymbol{x}^{(k)} = \boldsymbol{A}_{\boldsymbol{\theta}}^T \boldsymbol{\Sigma}_{\boldsymbol{\theta}}^{-1} \boldsymbol{y} + \boldsymbol{Q}_{\boldsymbol{\theta}} \boldsymbol{\mu} + \boldsymbol{v}_1 + \boldsymbol{v}_2, \tag{6.12}$$

with $\boldsymbol{v}_1 \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{A}_{\boldsymbol{\theta}}^T \boldsymbol{\Sigma}_{\boldsymbol{\theta}}^{-1} \boldsymbol{A}_{\boldsymbol{\theta}})$ and $\boldsymbol{v}_2 \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{Q}_{\boldsymbol{\theta}})$, representing independent Gaussian random variables [2, 19].

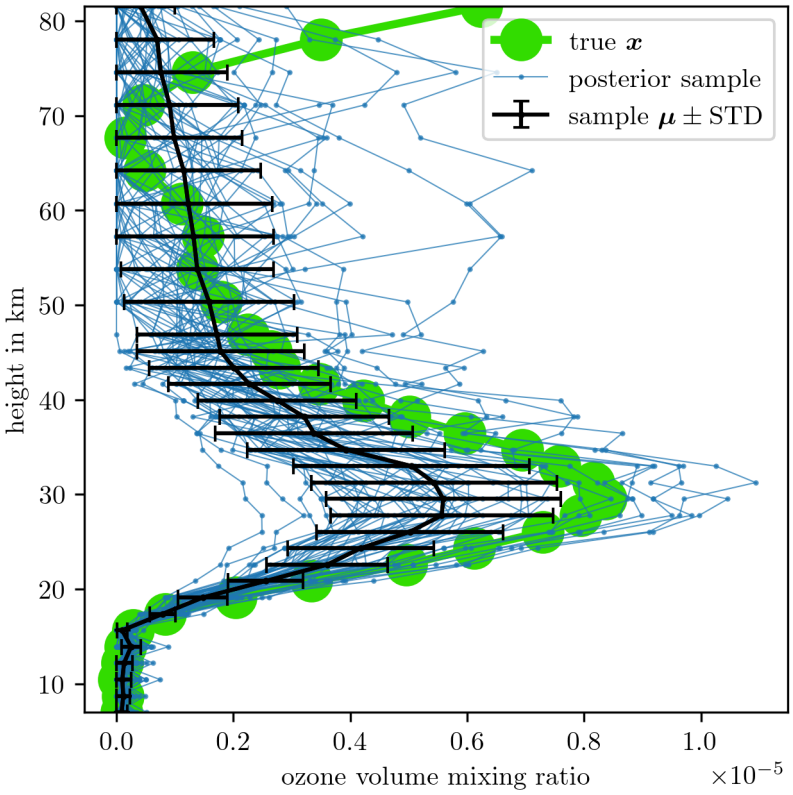


Figure 6.15: According to the hyper-parameter samples from the marginal posterior distribution (see Fig. 6.7) we plot the corresponding ozone sample from the full conditional posterior using the RTO method.

We observe that pressure and ozone are highly correlated. Since the hyper-parameter b is smaller than its ground truth value, the posterior pressure profile does not exponentially decrease as strongly as the ground truth pressure (see Fig. 6.7). This results in posterior pressure values which are slightly larger than the ground truth, and in an average posterior ozone profile with much smaller peak values compared to the ground truth. Additionally, the individual posterior samples are more prior-dominated through larger λ values (see Fig.6.7) and hence slightly smoother. Again, we are not able to recover the second peak at high altitudes. The posterior temperature profiles look (as expected) similar to the prior temperature profiles.

7 Summary and Outlook

In this chapter, we summarise the key results and conclusions of our work and provide an outlook for future research. We compare the Bayesian approach to a regularisation approach and elaborate on the differences between sampling-based methods and the TT approximation. Lastly, we situate our results within the broader context of atmospheric modelling and discuss the implications for the future development of an atmospheric limb sounder.

7.1 Regularisation Solution vs. Hierarchical Bayesian Approach

Using a regularisation approach, we need 200 solves of $\boldsymbol{x}_\lambda$ to obtain one solution of this inverse problem. In contrast, the hierarchical Bayesian approach involves 25 function evaluations of the marginal posterior and then 10100 samples from the approximated marginal posterior followed by 20 evaluations of $\boldsymbol{x}_\lambda$ and $\boldsymbol{B}_\lambda^{-1}$ to characterise the full posterior in $\approx$ 0.5s. Utilising a TT approximation (including finding the mode) to compute the full posterior mean and covariance takes $\approx$ 0.025s, requiring only 400 function evaluations to approximate $\pi(\lambda, \gamma|\boldsymbol{x})$ and is almost as fast as the regularisation approach ($\approx$ 0.015). Regardless, either method has a runtime of much less than a second on a basic laptop.

While regularisation yields a single optimal solution (point estimate), a Bayesian framework provides a distribution of ozone profiles, which are all feasible solutions to the inverse problem, and hence true errors. Moreover, within the hierarchical Bayesian approach, we can include prior knowledge about the noise, ozone profile and many more physical processes through hyper-parameters, offering an arbitrarily flexible and informative inference framework.

7.2 Sampling Methods vs. TT Approximation

Using the TT approximation involves far fewer function evaluations of the target distribution compared to sample-based methods, but requires a predefined grid and a normalisation constant, which, for now, we have to find iteratively. Relying solely on TT approximations may lead to a substantial amount of trial and error and dealing with numerical issues. Nevertheless, once properly configured, we have shown the potential and advantages of TT methods.

More specifically, the TT approximation of the 2-dimensional marginal posterior (≈ 0.02s) is more than 20 times faster than the MWG sampler (≈ 0.5s). Excluding the function evaluations for finding the mode of the marginal posterior, the MWG sampler takes 10100 steps while the TT approximation only needs 400 function evaluations; this is a factor of ≈ 25. Alternatively, for low-dimensional distributions, it may be preferable to approximate integrals directly using existing freely available quadrature libraries and packages such as `quadpy`.

In higher dimensions, such as the 18-dimensional marginal posterior considered in this thesis, TT methods (≈ 0.5min) outperform samplers like the t-walk (≈ 10min), once a grid and normalisation constant have been defined. Although the t-walk may not be the best sampler for this specific marginal posterior and the underlying correlation structure, it is robust and easy to implement. To illustrate the efficiency of TT approximations, we compare the number of function evaluations per 1000 independent samples. For 1000 independent samples with a maximum IACT of 550 and a burn-in period of 100 independent samples, the t-walk needs 1210000 function evaluations. In contrast, 24120 function evaluations are enough to approximate the marginal posterior in the TT format. Then drawing 1000 independent samples via the SIRT-MH scheme requires another 2000 function evaluations with an IACT of ≈ 1.2. So the cost per independent sample for the t-walk is 1210 and for the TT approach is ≈ 27 function evaluations, including the burn-in period and the TT approximation via the `rect_cross.cross` Python function. Or, after the burn-in phase, the t-walk requires around 1100 function evaluations per independent sample, while with an approximation of a probability density in the TT format available, only two function evaluations per independent sample are needed.

For future application, we suggest improving the efficiency of the TT approximation by, e.g., reducing the correlation structure through a coordinate system rotation or using better interpolators in between grid points to reduce the approximation error. This may be particularly important when the CDF in the SIRT scheme is not smooth due to poor approximations of the target density at previous samples. Moreover, using a different reference measure for integration as in [9], such as a Gaussian measure instead

of the current Lebesgue measure, may increase numerical stability. Currently, we have to predefine a normalisation constant and lower ranks manually, bounding the ranks automatically would be helpful (see e.g. [47]).

7.3 Atmospheric Physics

Here we summarise results within the context of our simplified physical atmospheric limb-sounding model. We demonstrated that the underlying non-linear forward model can be approximated with an affine map and the linear model, making this a linear inverse problem. For future application, we wish to include more measurement device-specific hyper-parameters in the forward model. This could include e.g. uncertainty in pointing accuracy or an antenna response function.

In Sec. 3.1, we showed that we do not gain more information if we measure more frequently or collect more data in noise-dominated regions and that we need an SNR of $\approx 10^7$ to produce data, which is informative about ozone at higher altitudes.

Fig. 6.15 and Fig. 6.14 illustrate that pressure and ozone are highly correlated. One has to consider that when conditioning on pressure estimates from other measurements, as a slight change in pressure does skew the ozone VMR significantly. We could fix that by choosing a more restrictive prior for the pressure-related hyper-parameter b, but that would not be objective. By explanatory analysis, we found that data with an SNR of ≈ 10000 recovers an ozone (without a peak in higher altitudes) and pressure profile close to the ground truth. As previously mentioned in the prior analysis (see Fig. 6.4), the model as well as the data are uninformative about temperature and dominated by the exponentially decreasing pressure.

All the samples plotted in Fig. 4.10, Fig. 4.5, and Fig. 6.15 present valid solutions to the inverse problem, but consistently fail to capture the ozone peak at higher altitudes. This is due to noise-dominated data (see Fig. 3.7) and low signal strength in upper atmospheric regions, where the variability of the posterior ozone is large and primarily determined by the prior. We conclude that the main objective for future research is to develop a more accurate, potentially parametrised (prior) model, which captures physical properties and chemical processes of ozone in the atmosphere.

References

[1] Ambrosio, Luigi, Brué, Elia, and Semola, Daniele. *Lectures on Optimal Transport*. Cham: Springer Nature Switzerland, 2024.

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

[3] Bardsley, Johnathan and Cui, Tiangang. "A Metropolis-Hastings-Within-Gibbs Sampler for Nonlinear Hierarchical-Bayesian Inverse Problems". In: *2017 MATRIX Annals*. Vol. 2. MATRIX Book Series. Switzerland: Springer, 2019, pp. 2–12.

[4] Bardsley, Johnathan 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.

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

[6] Champ, Charles W. and Sills, Andrew V. "The Generalized Law of Total Covariance". In: *preprint* (2022).

[7] Christen, J. Andrés and Fox, Colin. "A general purpose sampling algorithm for continuous distributions (the t-walk)". In: *Bayesian Analysis* 5.2 (2010), pp. 263 –281.

[8] Christen, J. Andrés and Fox, Colin. *The t-walk software*. `https://www.cimat.mx/~jac/twalk/`. [Online; accessed 25/11/24]. CIMAT, Mexico, and University of Otago, New Zealand.

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

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

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

[12] Dolgov, Sergey and Scheichl, Robert. "A Hybrid Alternating Least Squares–TT-Cross Algorithm for Parametric PDEs". In: *SIAM/ASA Journal on Uncertainty Quantification* 7.1 (2019), pp. 260–291.

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

[14] Duncan, Bryan. *Aura at 20 Years*. `https://science.nasa.gov/science-research/earth-science/aura-at-20-years/`. [Online; accessed 31/08/25]. NASA's Goddard Space Flight Center (GSFC), 2024.

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

[16] Feydy, Jean. "Analyse de données géométriques, au delà des convolutions". Ph.D. Thesis. Université Paris-Saclay, July 2020.

[17] Feydy, Jean. *GeomLoss – Geometric Loss functions between sampled measures, images and volumes*. `https://www.kernel-operations.io/geomloss/api/pytorch-api.html`. [Online; accessed 12/09/25].

[18] Fox, Colin. *Blokkurs on computing MCMC for inverse problems.* unpublished. Physics Department, University of Otago, 2025.

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

[20] Fox, Colin 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.

[21] Froidevaux, L. et al. "Validation of Aura Microwave Limb Sounder stratospheric ozone measurements". In: *Journal of Geophysical Research: Atmospheres* 113.D15 (2008).

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

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

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

[25] Hansen, Per Christian. "Regularization, GSVD and truncated GSVD". In: *BIT numerical mathematics* 29.3 (1989), pp. 491–504.

[26] Hansen, Per Christian. "The L-Curve and its Use in the Numerical Treatment of Inverse Problems". English. In: *Computational Inverse Problems in Electrocardiology.* Ed. by Johnston, P. WIT Press, 2001, pp. 119–142.

[27] Hansen, Per Christian and O'Leary, Dianne Prost. "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.

[28] Hesse, Drik. *py-uwerr; Python implementation of Monte Carlo error analysis a la Wolff.* `https://github.com/dhesse/py-uwerr`. [Online; accessed 09/09/25].

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

[30] Lee, Jae N. and Wu, Dong L. "Solar Cycle Modulation of Nighttime Ozone Near the Mesopause as Observed by MLS". In: *Earth and Space Science* 7.4 (2020).

[31] Livesey, Nathaniel J. et al. *Earth Observing System (EOS) Microwave Limb Sounder (MLS) Version 5.0x Level 2 and 3 data quality and description document.* Version 5.0-1.1a. NASA Goddard Earth Sciences Data and Information Services Center, 2022.

[32] Livesey, Nathaniel J et al. "Retrieval algorithms for the EOS Microwave limb sounder (MLS)". In: *IEEE Transactions on Geoscience and Remote Sensing* 44.5 (2006), pp. 1144–1155.

[33] Livesey, Nathaniel J. et al. "Validation of Aura Microwave Limb Sounder O3 and CO observations in the upper troposphere and lower stratosphere". In: *Journal of Geophysical Research: Atmospheres* 113.D15 (2008).

[34] M., Schwartz 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]. NASA Goddard Earth Sciences Data and Information Services Center, 2020.

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

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

[37] Oseledets, Ivan. "DMRG Approach to Fast Linear Algebra in the TT-Format". In: *Computational Methods in Applied Mathematics* 11.3 (2011), pp. 382–393.

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

[39] Oseledets, Ivan, Bershatsky, Daniel, and Saluev, Tigran. *ttpy - a Python implementation of the TT-toolbox*. `https://github.com/oseledets/ttpy`. [Online; accessed 23/06/25]. 2018.

[40] Oseledets, Ivan and Tyrtyshnikov, Eugene. “TT-cross approximation for multidimensional arrays”. In: *Linear Algebra and its Applications* 432.1 (2010), pp. 70–88.

[41] Read, W.G. et al. “The clear-sky unpolarized forward model for the EOS aura microwave limb sounder (MLS)”. In: *IEEE Transactions on Geoscience and Remote Sensing* 44.5 (2006), pp. 1367–1379.

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

[43] Roberts, Gareth. *ST911 Fundamentals of Statistical Inference - Part III*. Department of Statistics, University of Warwick, 2015.

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

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

[46] Rodgers, Clive D. “Retrieval of atmospheric temperature and composition from remote measurements of thermal radiation”. In: *Reviews of Geophysics* 14.4 (1976), pp. 609–624.

[47] Rohrbach, Paul B. et al. “Rank Bounds for Approximating Gaussian Densities in the Tensor-Train Format”. In: *SIAM/ASA Journal on Uncertainty Quantification* 10.3 (2022), pp. 1191–1224.

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

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

[50] Santosh, KC, Das, Nibaran, and Ghosh, Swarnendu. “Chapter 3 - Deep learning models”. In: *Deep Learning Models for Medical Imaging*. Primers in Biomedical Imaging Devices and Systems. Academic Press, 2022, pp. 65–97.

[51] Satopäa, Ville 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.

[52] Sedlmeir, Florian et al. “Detecting THz in the telecom range: All resonant THz up-conversion in a whispering gallery mode resonator”. In: *2014 39th International Conference on Infrared, Millimeter, and Terahertz waves (IRMMW-THz)*. 2014, pp. 1–2.

[53] Šimečková, Marie 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.

[54] Simpson, Daniel, Lindgren, Finn, and Rue, Håvard. “Think continuous: Markovian Gaussian models in spatial statistics”. In: *Spatial Statistics* 1 (2012), pp. 16–29.

[55] Sokal, Alan. “Monte Carlo Methods in Statistical Mechanics: Foundations and New Algorithms”. In: *Functional Integration: Basics and Applications*. Boston, MA: Springer US, 1997, pp. 131–192.

[56] Suresh, Mallika Irene et al. “Multichannel upconversion of terahertz radiation in an optical disk resonator”. In: *Opt. Express* 33.5 (2025), pp. 10302–10311.

[57] Tan, Sze M, Fox, Colin, and Nicholls, Geoff K. *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]. Physics Department, University of Otago, 2016.

[58] Thickstun, John. *Kantorovich-rubinstein duality.* `https://courses.cs.washington.edu/courses/cse599i/20au/resources/L12_duality.pdf`. [Online; accessed 31/08/25]. University of Washington, 2019.

[59] *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.

[60] Ustin, Susan and Middleton, Elizabeth McPhee. "Current and near-term Earth-observing environmental satellites, their missions, characteristics, instruments, and applications". In: *Sensors* 24.11 (2024), p. 3488.

[61] Vats, Dootika et al. "Understanding Linchpin Variables in Markov Chain Monte Carlo". In: *preprint* (2022).

[62] Wang, Yu-Xiang et al. "Trend Filtering on Graphs". In: *Journal of Machine Learning Research* 17.105 (2016), pp. 1–41.

[63] Waters, Joe et al. "The earth observing system microwave limb sounder (EOS MLS) on the Aura satellite". In: *IEEE Transactions on Geoscience and Remote Sensing* 44.5 (2006), pp. 1075–1092.

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

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

[66] Wolff, Ulli et al. *Lecture Notes on Computational Physics II [in german].* `www-com.physik.hu-berlin.de/comphys/comphys.htm`. [Online; accessed 29/08/25]. Humboldt University, Berlin, 2016.

Appendices

A

Theoretical and Technical Background

A.1 Correlation Structure

In the book Gaussian Markov Random Fields [48], Rue and Held demonstrate that a strong correlation between the hyper-parameter μ and the latent field $\boldsymbol{x}$ can significantly slow down convergence particularly when using Gibbs samplers. They consider the hierarchical model

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

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

and apply a Gibbs sampler based on the full conditional distributions

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

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

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

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

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

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

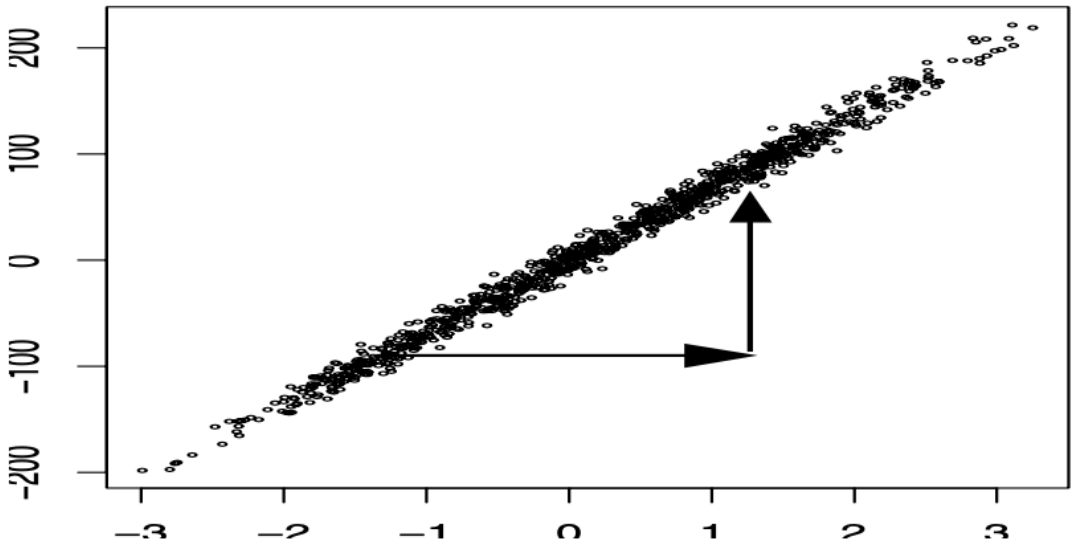


Figure A.1: The figure taken from [48, Figure 4.1 (b)] shows samples from the chain of μ (x-axis) and $\mathbf{1}^T \boldsymbol{Q}\boldsymbol{x}^{(k)}$ (y-axis) for over 1000 iterations, based on the hierarchical model in Eq. A.1, with an autoregressive process encoded in $\boldsymbol{Q}$. The algorithm updates μ and $\boldsymbol{x}$ successively from their conditional distributions (see Eq. A.2 and Eq. A.3). 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.

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.

A.2 Monte-Carlo Error and Integrated Autocorrelation Time

To assess the error $(\sigma^{(i)})^2$ of a samples-based estimate

$$\bar{h}_N^{(i)} := \mathrm{E}_{\boldsymbol{x}|\boldsymbol{y}}[h(\boldsymbol{x})] = \frac{1}{N}\sum_{k=1}^{N} h(\boldsymbol{x}^{(k)}), \tag{A.6}$$

from the chain $\mathcal{M}^{(i)} = \{\boldsymbol{x}^{(1)}, \ldots, \boldsymbol{x}^{(k)}, \ldots, \boldsymbol{x}^{(s)}, \ldots, \boldsymbol{x}^{(N)}\} \sim \pi(\boldsymbol{x}|\boldsymbol{y})$, we ignore systematic error due to initialisation bias (burn-in period), but we have to take into account that samples produced by any system or algorithm are correlated. To derive the IACT, we follow Ulli Wolff's lecture notes [66] (or alternatively [64]).

In general, the error of a Monte-Carlo estimate is:

$$(\sigma^{(i)})^2 = \mathrm{Var}(\bar{h}_N^{(i)}) = \mathrm{Var}(\mathrm{E}_{\boldsymbol{x}|\boldsymbol{y}}[h(\boldsymbol{x})]) = \left(\frac{1}{N}\sum_{k=1}^{N} h(\boldsymbol{x}^{(k)}) - \bar{h}_N^{(i)}\right)^2. \tag{A.7}$$

Expanding this summation, we see that

$$(\sigma^{(i)})^2 = \frac{1}{N^2} \sum_{k,s=1}^{N} \Gamma(k-s) \tag{A.8}$$

with the autocorrelation coefficient $\Gamma(k-s) = \big(h(\boldsymbol{x}^{(k)}) - \bar{h}_N^{(i)}\big)\big(h(\boldsymbol{x}^{(s)}) - \bar{h}_N^{(i)}\big)$. Next we rewrite

$$\sum_{k,s=1}^{N} \Gamma(k-s) = \mathrm{Var}(h(\boldsymbol{x})) \sum_{k,s=1}^{N} \frac{\Gamma(k-s)}{\Gamma(0)} = \mathrm{Var}(h(\boldsymbol{x})) \sum_{k,s=1}^{N} \rho(k-s)\,, \tag{A.9}$$

with the normalised autocorrelation coefficient $\rho(k-s) = \Gamma(k-s)/\Gamma(0)$ at lag $k-s$ and $\Gamma(0) = \mathrm{Var}(h(\boldsymbol{x}))$ for $k = s$. Typically $\Gamma(t)$ decays exponentially so that, for $N \gg \tau$, $\Gamma(t) \overset{t\to\infty}{\propto} \exp\{-|t|/\tau\}$ and we can approximate

$$\sum_{k,s=1}^{N} \rho(k-s) = N \sum_{t=-(N-1)}^{N-1} \left(1 - \frac{t}{N}\right)\rho(t) \approx N \sum_{t=-\infty}^{\infty} \rho(t)\,, \tag{A.10}$$

see [55, p. 137]. If $\tau \gg 1$

$$\sum_{t=-\infty}^{\infty} \rho(t) = 1 + 2\sum_{t=1}^{\infty} \left(e^{-1/\tau}\right)^t = 1 + 2\frac{e^{-1/\tau}}{1 - e^{-1/\tau}} \approx 1 + 2\frac{1 - 1/\tau}{1/\tau} = 2\tau - 1\,. \tag{A.11}$$

Here we use the geometric power series $\sum_{n=0}^{\infty} x^n = 1/(1-x)$ and the Taylor series $e^x \approx 1 + x$ for small x. In practice, the estimate for the Monte-Carlo error is:

$$(\sigma^{(i)})^2 \approx \frac{\mathrm{Var}(h(\boldsymbol{x}))}{N} \sum_{t=-\infty}^{\infty} \rho(t) \approx \frac{\mathrm{Var}(h(\boldsymbol{x}))}{N} \underbrace{\left(1 + 2\sum_{t=1}^{W} \rho(t)\right)}_{:=\tau_{\mathrm{int}}} = \mathrm{Var}(h(\boldsymbol{x}))\frac{\tau_{\mathrm{int}}}{N}\,, \tag{A.12}$$

where W is the summation window and we define the IACT as twice the value as in [66, pp. 103-105]. The IACT provides a good estimate of how many steps the sampling algorithm needs to take to produce one independent sample. More specifically, the effective sample size $\frac{\tau_{\mathrm{int}}}{N}$ gives an estimate of how efficient a sampler is.

A.3 Python Code

```python
def MargBack(TTCore, univarGrid):
  ''' Backward marginalisation (see Prop.  1) as in SIRT from Cui et al. [9] '''

  dim = len(univarGrid)
  B = dim * [None]  # coeffTensor
  B[-1] = TTCore[-1]
  R = [None] * dim
  C = [None] * dim

  for k in range(dim - 1, 0, -1):
    r_kmin1, n, r_k = np.shape(TTCore[k])
    # Eq.  2.28, [9, Eq.  22] !! we set Lebesgue Measure to const = one
    M = np.identity(n) * (univarGrid[k][-1] - univarGrid[k][0])  # Mass matrix
    L = scy.linalg.cholesky(M)

    # construct Tensor C  Eq.  2.33, [9, Eq.  27]
    C[k] = np.zeros((r_kmin1, n, r_k))
    for alpha in range(0, r_kmin1):
      for l in range(0, r_k):
        C[k][alpha, :, l] = B[k][alpha, :, l] @ L[:, :]

    # unfold along first coordinate and compute thin QR decomposition of C^T
    #  Eq.  2.34, [9, Eq.  28]
    Q, R[k] = np.linalg.qr(C[k].reshape((r_kmin1, n * r_k), order='C').transpose(), mode='reduced')

    # compute next coefficient tensor Eq.  2.35, [9, Eq.  29]
    r_kmin2, n, r_kmin1 = np.shape(TTCore[k - 1])
    B[k - 1] = np.zeros(np.shape(TTCore[k - 1]))
    for alpha_2 in range(0, r_kmin2):
      for l_1 in range(0, r_kmin1):
        B[k - 1][alpha_2, :, l_1] = TTCore[k - 1][alpha_2, :, :] @ R[k][l_1, :]

return B
```

Listing A.1: Python code to calculate Backward marginals, as in Prop. 1 and [9].

```python
def MargForw(TTCore, univarGrid):
''' Forward marginalisation (see Prop. 2)
  similar to backward marginalsiation as in Cui et al. [9] '''

# compute pre marginal coefficients sarting at dim = 1, k = 0
BPre = dim * [None]  # coeffTensor
LebLam = 1  # !! Lebesgue Measure
BPre[0] = TTCore[0]
RPre = [None] * dim
CPre = [None] * dim

for k in range(0, dim-1):
  r_kmin1, n, r_k = np.shape(TTCore[k])
  #  Eq. 2.28, [9, Eq. 22] !! we set Lebesgue Measure to const = one
  M = np.identity(n) * (univarGrid[k][-1] - univarGrid[k][0])  # Mass matrix
  L = scy.linalg.cholesky(M)

  # construct Tensor C  Eq. 2.36
  CPre[k] = np.zeros((r_kmin1, n, r_k))
  for alpha in range(0, r_kmin1):
    for l in range(0, r_k):
      CPre[k][alpha, :, l] = BPre[k][alpha, :, l] @ L[:, :]

  # unfold along first coordinate and compute thin QR decomposition of C
  # Eq. 2.37
  Q, RPre[k] = np.linalg.qr(CPre[k].reshape((r_kmin1 * n, r_k), order='C'), mode='reduced')

  # compute next coefficient tensor  Eq. 2.38
  r_k, n, r_kpls1 = np.shape(TTCore[k + 1])
  BPre[k + 1] = np.zeros(np.shape(TTCore[k + 1]))
  for alpha_1 in range(0, r_kpls1):
    for l_1 in range(0, r_k):
      BPre[k + 1][l_1, :, alpha_1] = RPre[k][l_1,:]
                                          @ TTCore[k + 1][:, :, alpha_1]

return BPre
```

Listing A.2: Python code to calculate forward marginals, as in Prop. 2.

```python
def SIRT(seeds, SQTT, univarGrid, BackMarg, absError):
  ''' do squared inverse rosenblatt transform (SIRT) as in Cui et al. [9] '''

  dim, numbSampl = seeds.shape
  sampls = np.zeros(seeds.shape) # samples from approximated PDF
  probVal = np.zeros(seeds.shape) # PDF values, for MH-correction step
  Approx = np.zeros(seeds.shape[1]) # TT-Approx., to compare to true function

  # Lebesgue measure for quadrature Eq.  2.20
  WholeLebLam = np.zeros(dim)
  for k in range(0, dim):
    WholeLebLam[k] = (univarGrid[k][-1] - univarGrid[k][0])
  lamX = np.ones(dim)
  for k in range(1, dim):
    lamX[k - 1] = np.prod(WholeLebLam[k:])


  gamError = absError / np.prod(WholeLebLam) # error as in Eq.  2.25 [9, Eq.  (35)]

  # sample from first dimension [9, Eq.  (30)]
  firstMarg = gamError * lamX[0] + np.sum(BackMarg[0][0, :, :] ** 2, 1)
  # cumulative distribution function, normalised numerically Eq.  2.39 [9, Eq.  (17)]
  firstCDF = np.cumsum(firstMarg / np.sum(firstMarg))
  # draw samples as 'inverse transform'
  sampls[0] = np.interp(seeds[0], firstCDF, univarGrid[0])
  probVal[0] = np.interp(sampls[0], univarGrid[0], firstMarg / np.sum(firstMarg))

  # sample from other dimensions
  for n in range(0, numbSampl):
    # piecew. poly. interpol in first dimension Eq.  2.41 [13]
    CurrApprCore = LinInterPolTT(SQTT[0], univarGrid[0], sampls[0][n])
    for d in range(1, dim):
      # marginal function, conditioned on previous samples
      rank_min, gridSize, rank_pls = BackMarg[d].shape
      MargDep = np.zeros((BackMarg[d].shape))
      for r in range(0, rank_min):
        # condition on previous samples
        MargDep[r, :, :] = CurrApprCore[0, r] * BackMarg[d][r, :, :]

      currMarg = gamError * lamX[d] + np.sum(np.sum(np.copy(MargDep), axis=0)** 2,
        axis=1) # Eq. 2.40 [9, Eq.  (31)]

      currCDF = np.cumsum(currMarg / np.sum(currMarg)) # Eq.  2.39 [9, Eq.  (17)]

      # draw sample as 'inverse transform'
      sampls[d][n] = np.interp(seeds[d][n], currCDF, univarGrid[d])
      probVal[d][n] = np.interp(sampls[d][n], univarGrid[d],
        currMarg / np.sum(currMarg))
      # piecew. poly. interpol., Eq.  2.41 [13], cond. on sampl. for next PDF
      CurrApprCore = np.copy(CurrApprCore) @ LinInterPolTT(SQTT[d], univarGrid[d],
        sampls[d][n])

  Approx[n] = gamError + CurrApprCore ** 2

  return sampls, probVal, Approx
```

Listing A.3: Python code to draw samples via SIRT, as in Alg. Box 1.

B
Additional Figures

Some of the additional figures shown are repetitive and omitted from the main document; others provide details that are not necessary for understanding the main results but may be interesting and offer a more visual understanding for the curious reader.

B.1 Ozone

B.1.1 Ozone Prior

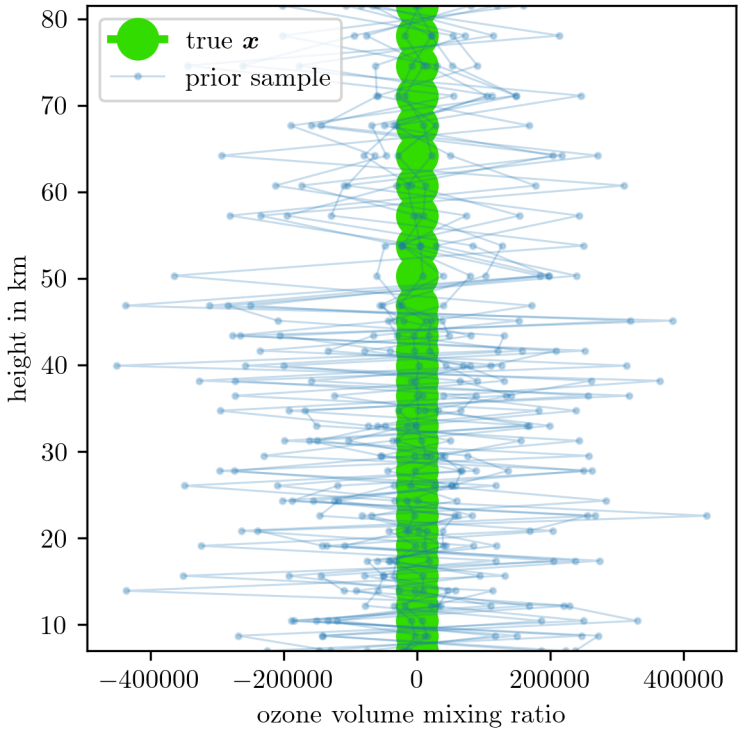


Figure B.1: We draw samples from ozone prior distribution $\boldsymbol{x} \sim \mathcal{N}(0, \delta \boldsymbol{L})$ after generating a sample from the hyper-prior distribution $\delta \sim \mathcal{T}(1, 10^{-10})$. Note that since the variance of prior samples is very large compared to the ozone volume mixing ratios, the ozone profile appears to be constant, which is not the case, see e.g. Fig. 4.5.

B.1.2 Integrated Autocorrelation Time

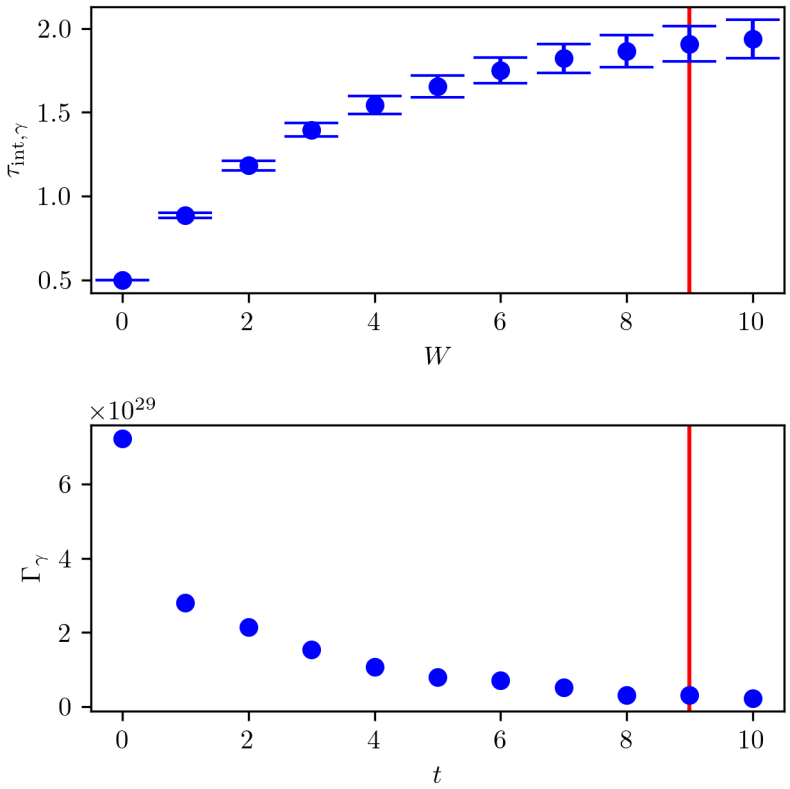


Figure B.2: Provided by [28], the IACT $\tau_{\text{int},\gamma}$ at summation windows W as well as the estimated autocorrelation function Γ_γ at lag t of the samples $\gamma \sim \pi(\cdot|\boldsymbol{y})$ based on the linear forward model.

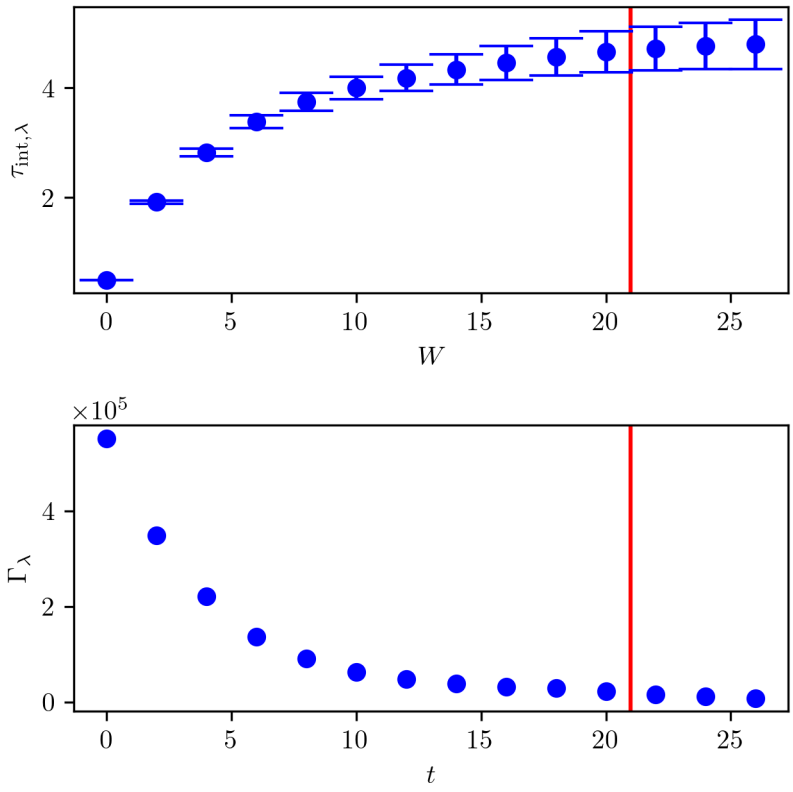


Figure B.3: Provided by [28], the IACT $\tau_{\text{int},\lambda}$ at summation windows W as well as the estimated autocorrelation function Γ_λ at lag t of the samples $\lambda \sim \pi(\cdot|\boldsymbol{y})$ based on the approximated forward model.

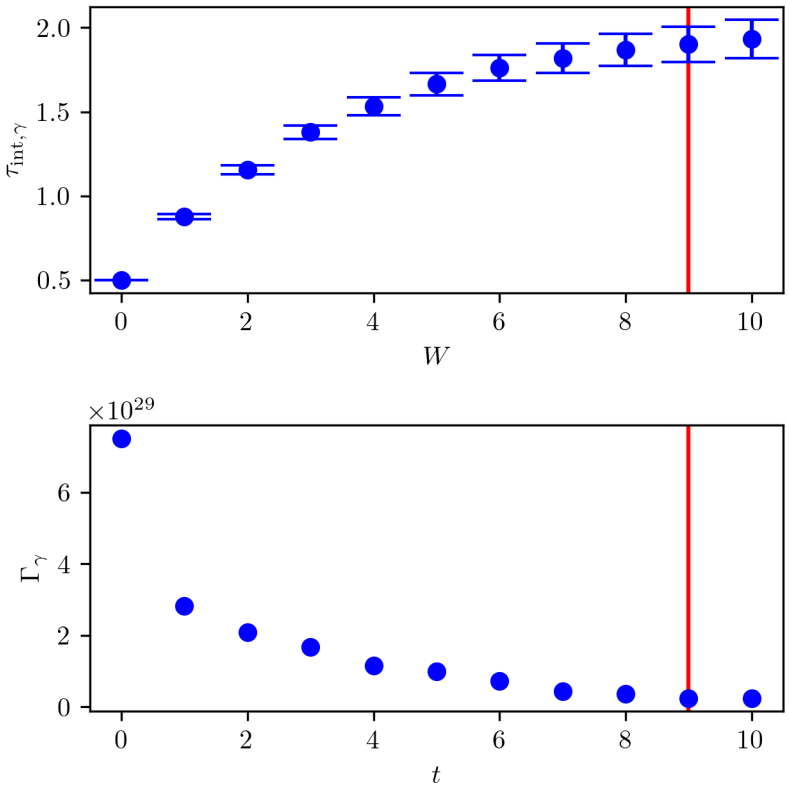


Figure B.4: Provided by [28], the IACT $\tau_{\text{int},\gamma}$ at summation windows W as well as the estimated autocorrelation function Γ_γ at lag t of the samples $\gamma \sim \pi(\cdot|\boldsymbol{y})$ based on the approximated forward model.

B.1.3 Eigenvectors of Full Conditional Posterior Precision Matrix

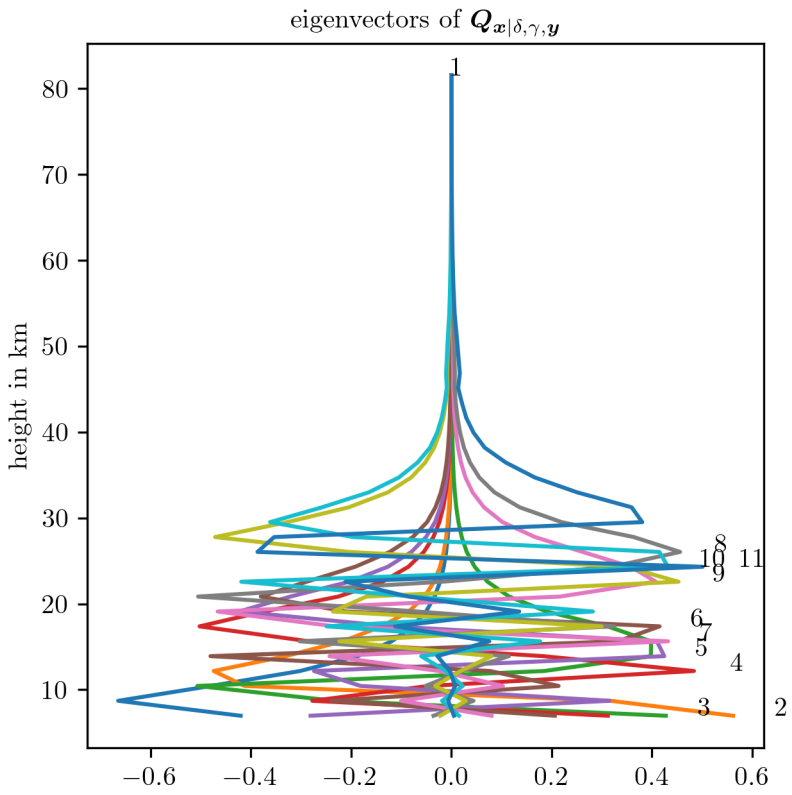


Figure B.5: First 11 eigenvectors corresponding to in size ordered eigenvalues of conditional precision matrix $\boldsymbol{Q}_{\boldsymbol{x}|\delta,\gamma,\boldsymbol{y}}$. We see that the eigenvectors span structures for heights ≤ 40.

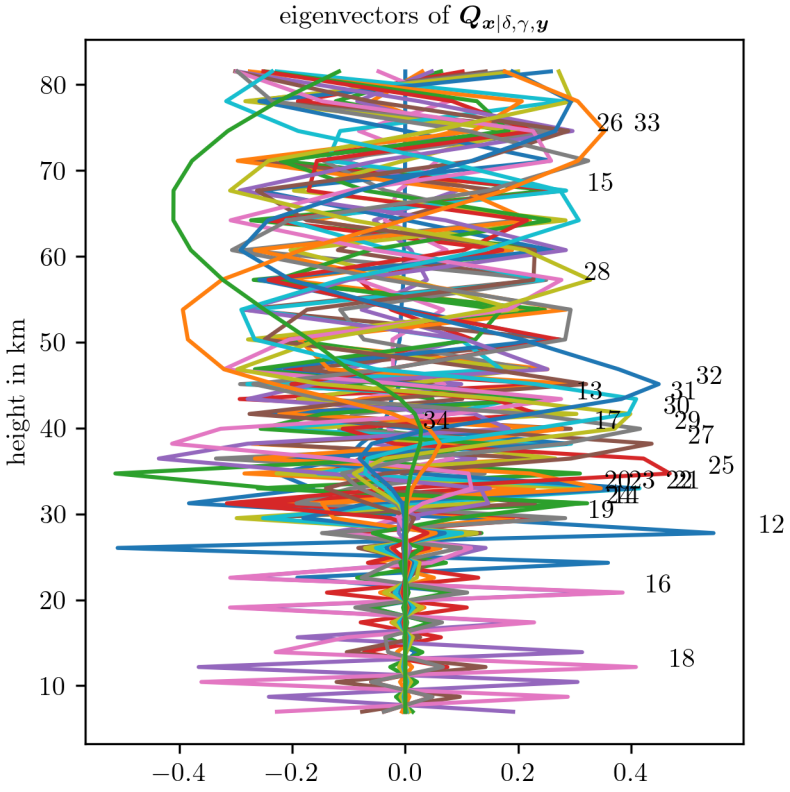


Figure B.6: Last 23 eigenvectors corresponding to in size ordered eigenvalues of conditional precision matrix $\boldsymbol{Q}_{\boldsymbol{x}|\delta,\gamma,\boldsymbol{y}}$. The eigenvectors represent structures according to the prior.

B.2 Pressure and Temperature

B.2.1 Priors

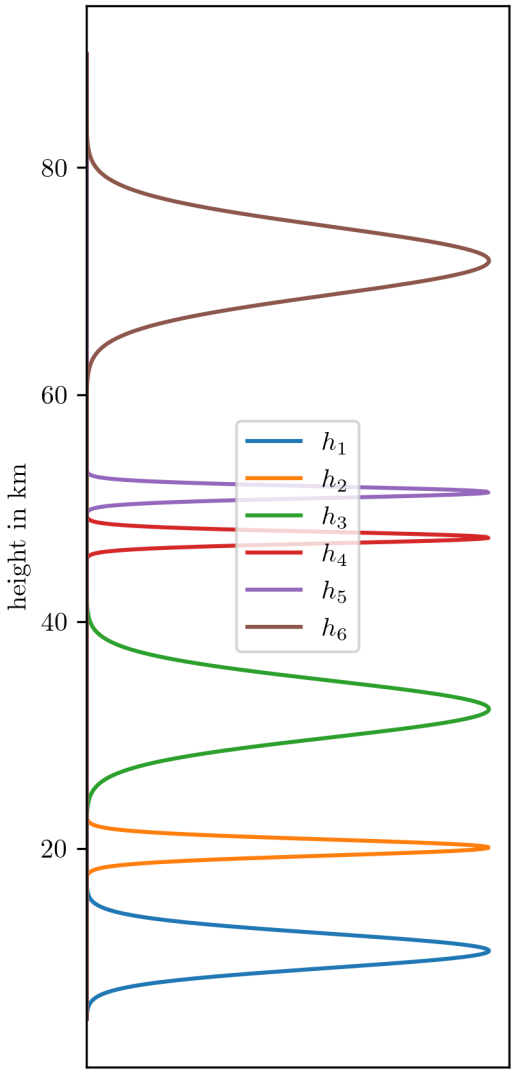


Figure B.7: Prior distributions $\pi(\boldsymbol{h_T})$, which we choose so that they do not overlap and not conflict with the temperature function in Eq. 3.11.

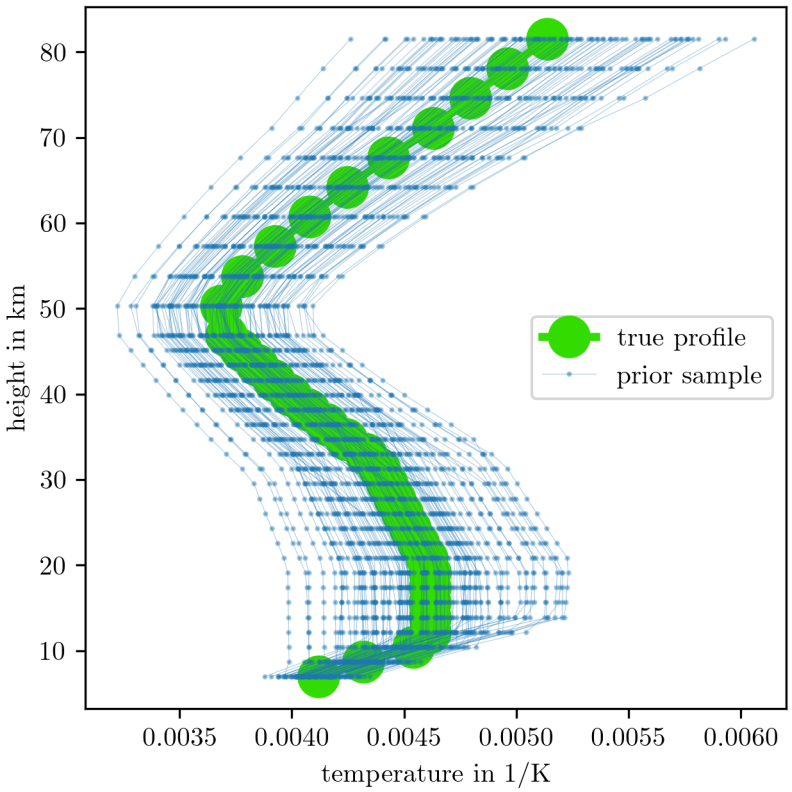


Figure B.8: Prior samples of the inverted temperature profile.

B.2.2 T-walk Trace

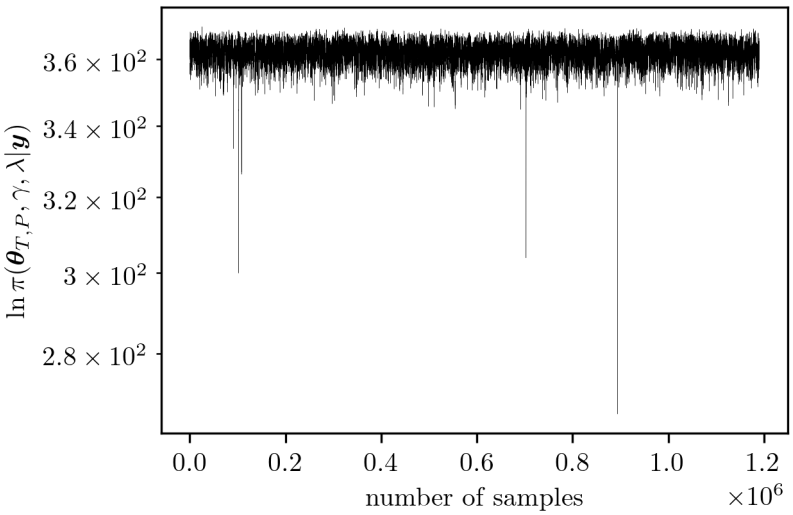


Figure B.9: Output trace of the t-walk on the posterior distribution $\pi(p_0, b, \boldsymbol{h_T}, \boldsymbol{a} | \gamma, \boldsymbol{y})$.

B.2.3 Integrated Autocorrelation Time

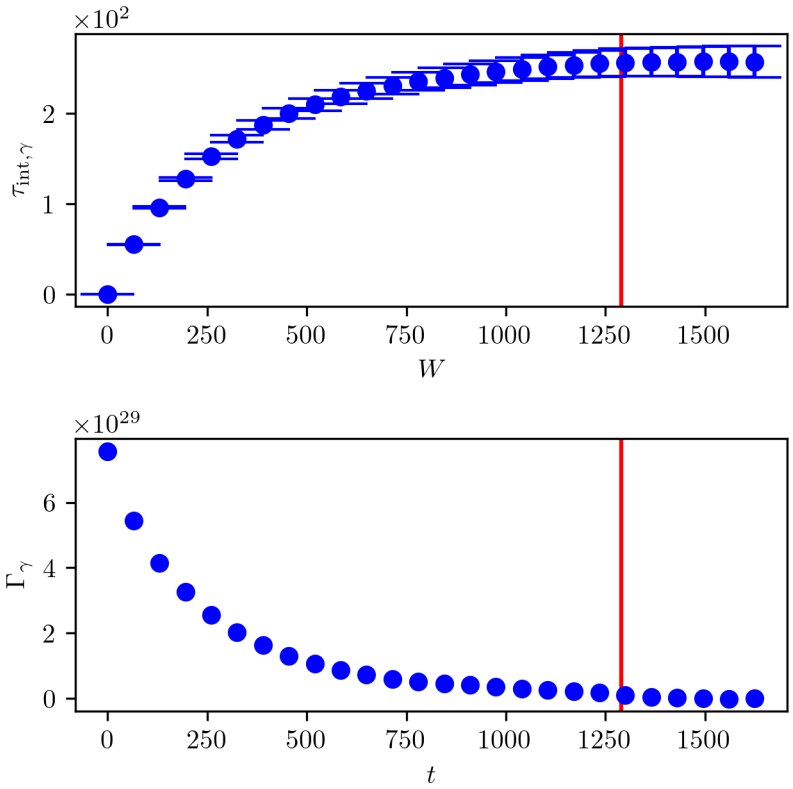


Figure B.10: Provided by [28], the IACT $\tau_{\text{int},\gamma}$ at summation windows W and the estimated autocorrelation function Γ_γ at lag t of samples $\gamma \sim \pi(\cdot|\boldsymbol{y})$ from the t-walk for the approximated forward model.

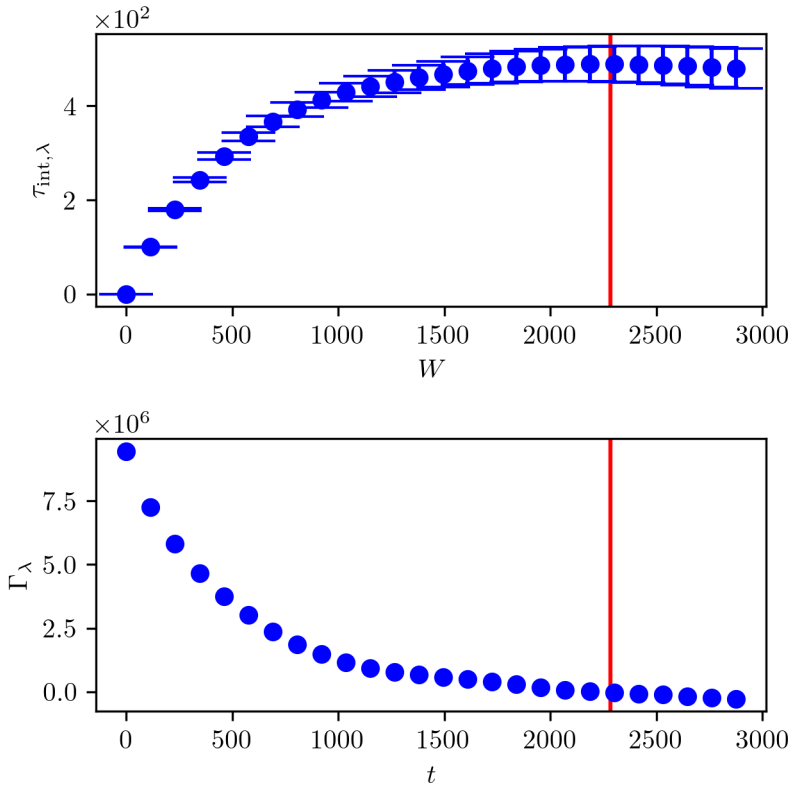


Figure B.11: Provided by [28], the IACT $\tau_{\text{int},\lambda}$ at summation windows W and the estimated autocorrelation function Γ_λ at lag t of samples $\lambda \sim \pi(\cdots|\boldsymbol{y})$ from the t-walkfor the approximated forward model.

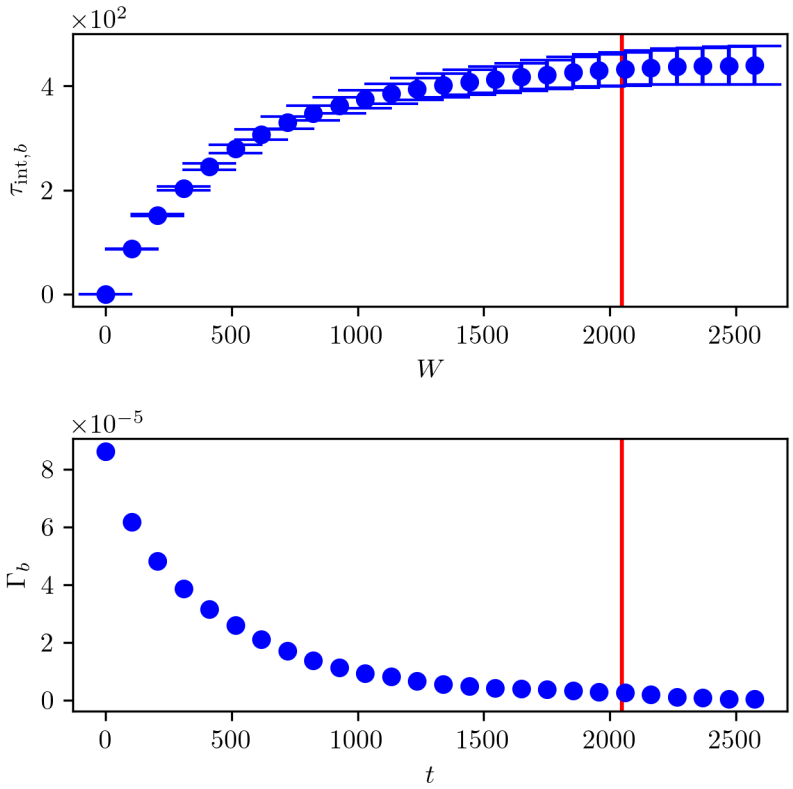


Figure B.12: Provided by [28], the IACT $\tau_{\text{int},b}$ at summation windows W and the estimated autocorrelation function Γ_b at lag t of samples $b \sim \pi(\cdot|\boldsymbol{y})$ from the t-walk for the approximated forward model.

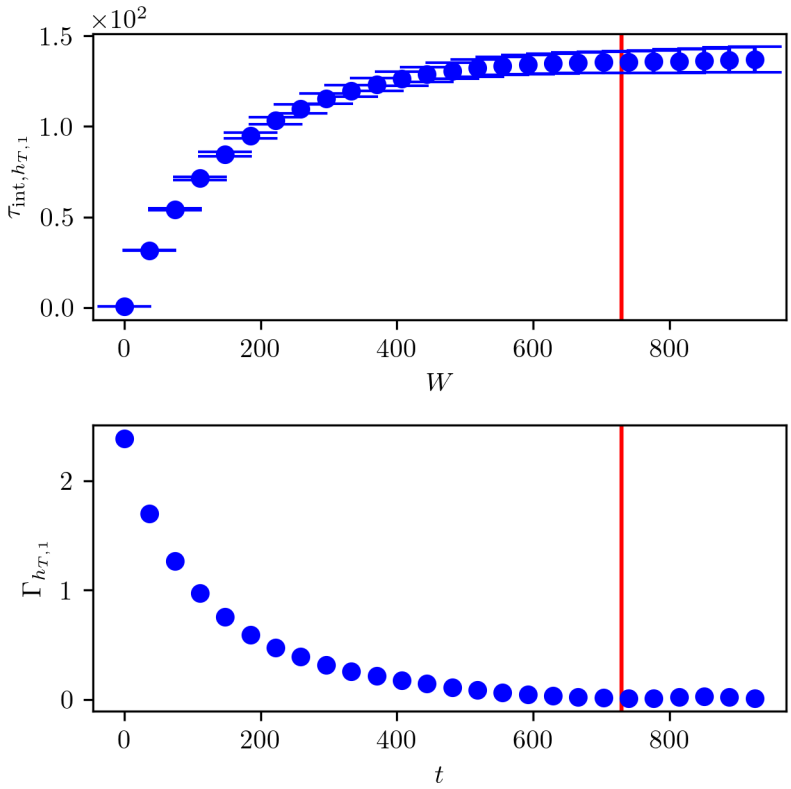


Figure B.13: Provided by [28], the IACT $\tau_{\mathrm{int},h_{T,1}}$ at summation windows W and the estimated autocorrelation function $\Gamma_{h_{T,1}}$ at lag t of samples $h_{T,1} \sim \pi(\cdot|\boldsymbol{y})$ from the t-walk for the approximated forward model.

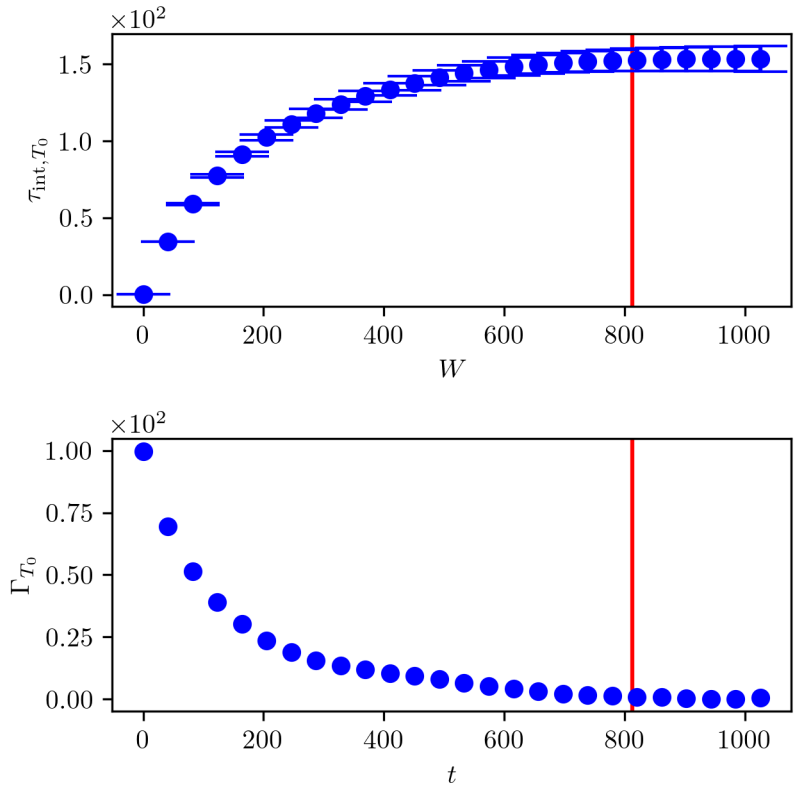


Figure B.14: Provided by [28], the IACT τ_{int,T_0} at summation windows W and the estimated autocorrelation function Γ_{T_0} at lag t of samples $T_0 \sim \pi(\cdot|\boldsymbol{y})$ from the t-walk for the approximated forward model.

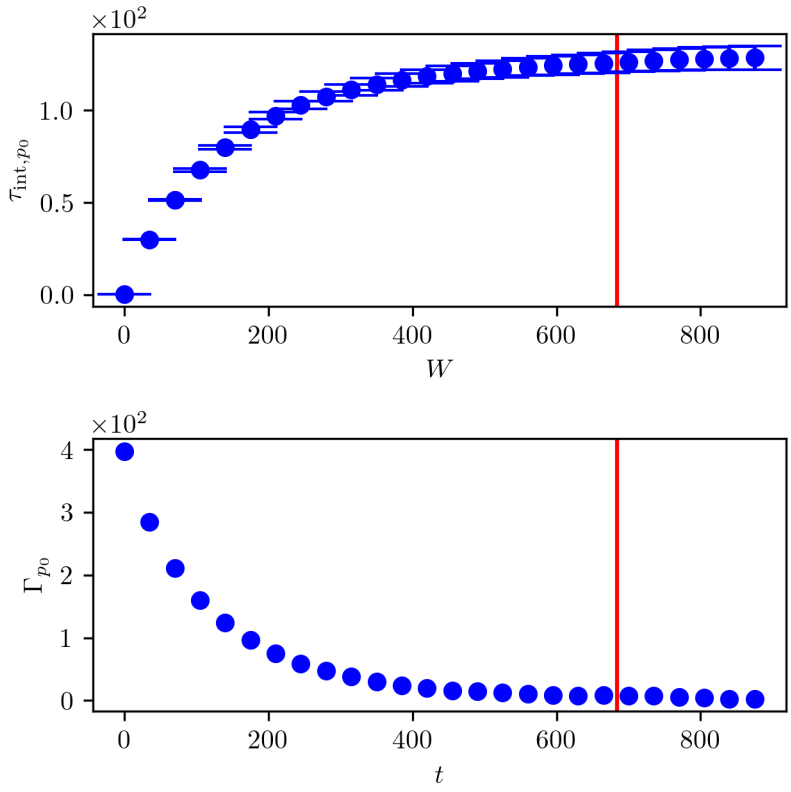


Figure B.15: Provided by [28], the IACT τ_{int,p_0} at summation windows W and the estimated autocorrelation function Γ_{p_0} at lag t of samples $p_0 \sim \pi(\cdot|\boldsymbol{y})$ from the t-walk for the approximated forward model.

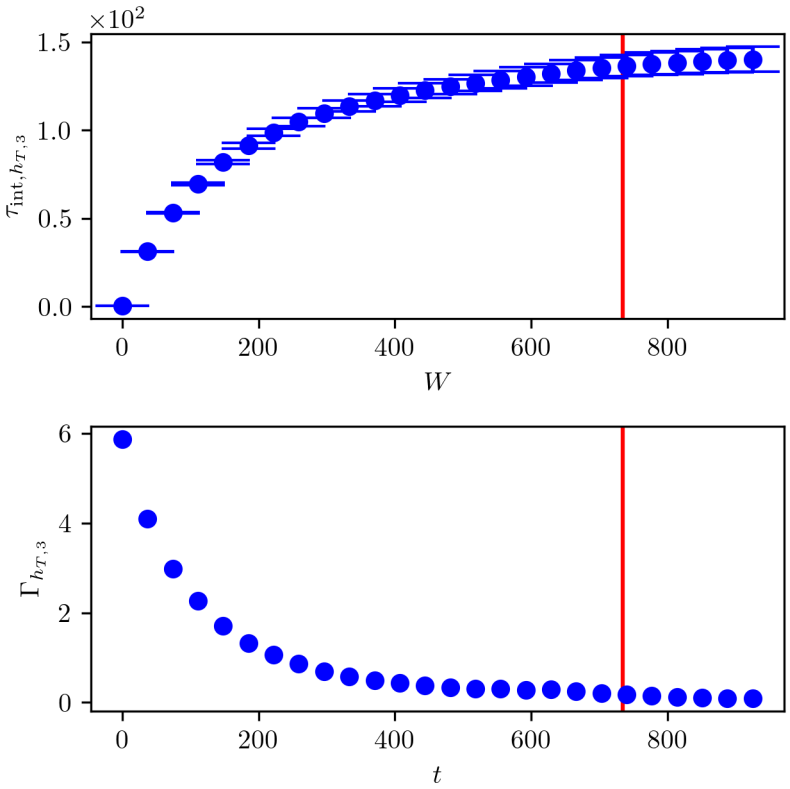


Figure B.16: Provided by [28], the IACT $\tau_{\text{int},h_{T,3}}$ at summation windows W and the estimated autocorrelation function $\Gamma_{h_{T,3}}$ at lag t of samples $h_{T,3} \sim \pi(\cdot|\boldsymbol{y})$ from the t-walk for the approximated forward model.

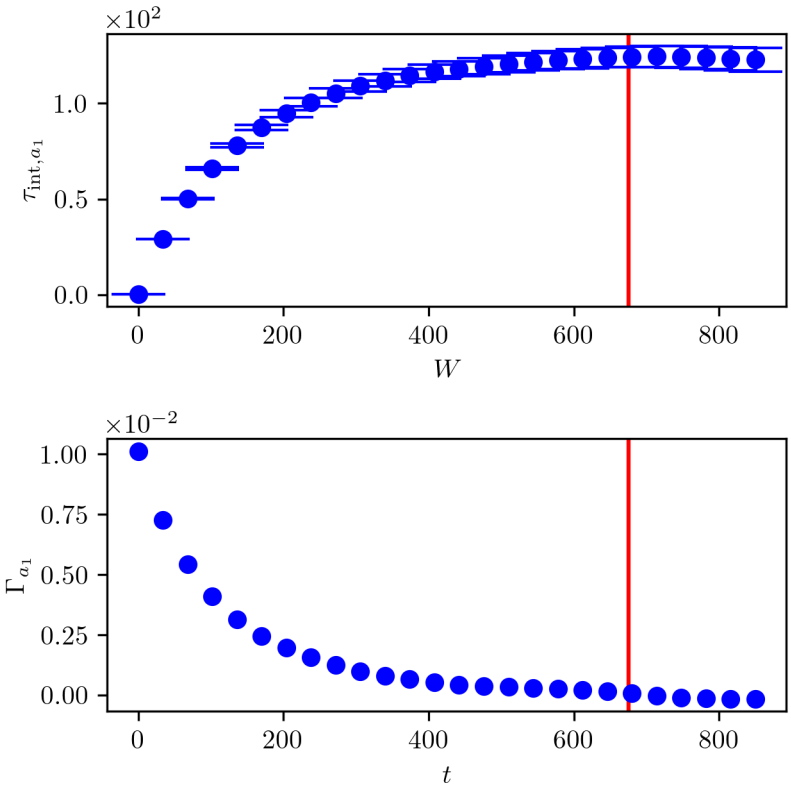


Figure B.17: Provided by [28], the IACT τ_{int,a_1} at summation windows W and the estimated autocorrelation function Γ_{a_1} at lag t of samples $a_1 \sim \pi(\cdot|\boldsymbol{y})$ from the t-walk for the approximated forward model.

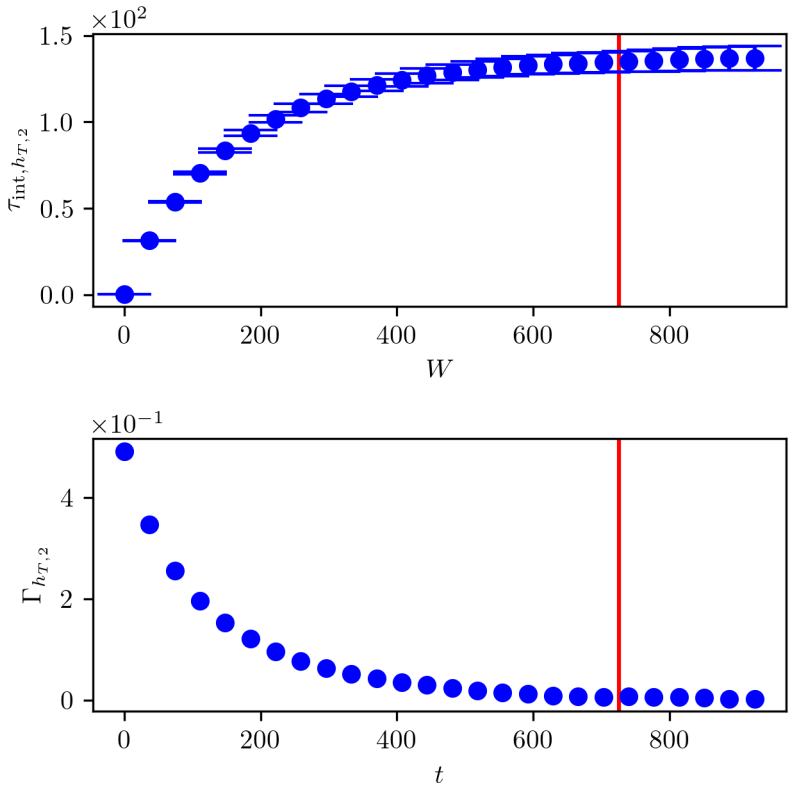


Figure B.18: Provided by [28], the IACT $\tau_{\mathrm{int},h_{T,2}}$ at summation windows W and the estimated autocorrelation function $\Gamma_{h_{T,2}}$ at lag t of samples $h_{T,2} \sim \pi(\cdots|\boldsymbol{y})$ from the t-walk for the approximated forward model.

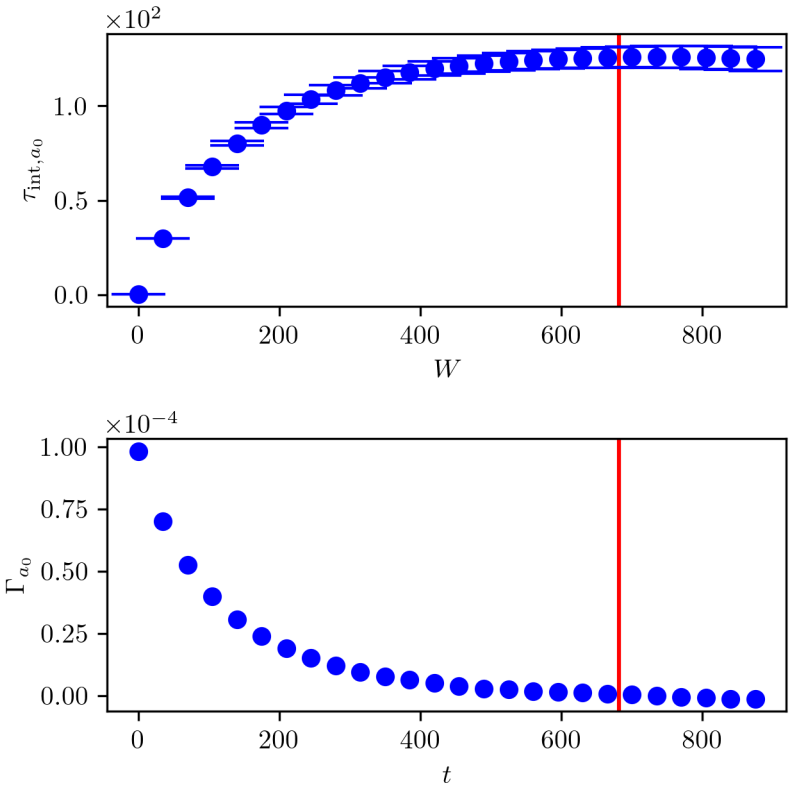


Figure B.19: Provided by [28], the IACT τ_{int,a_0} at summation windows W and the estimated autocorrelation function Γ_{a_0} at lag t of samples $a_0 \sim \pi(\cdot|\boldsymbol{y})$ from the t-walk for the approximated forward model.

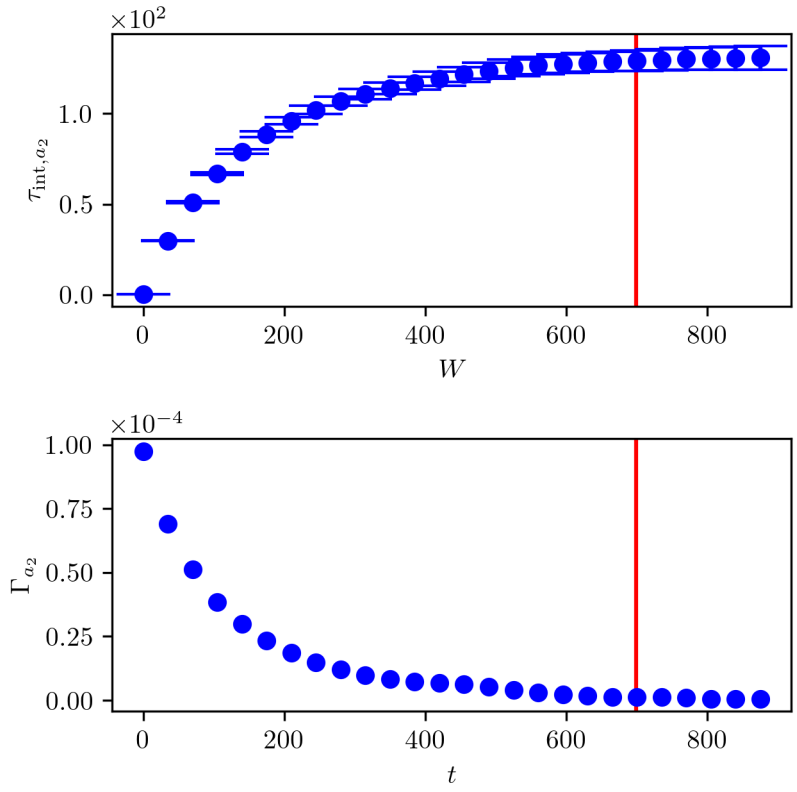


Figure B.20: Provided by [28], the IACT τ_{int,a_2} at summation windows W and the estimated autocorrelation function Γ_{a_2} at lag t of samples $a_2 \sim \pi(\cdot|\boldsymbol{y})$ from the t-walk for the approximated forward model.

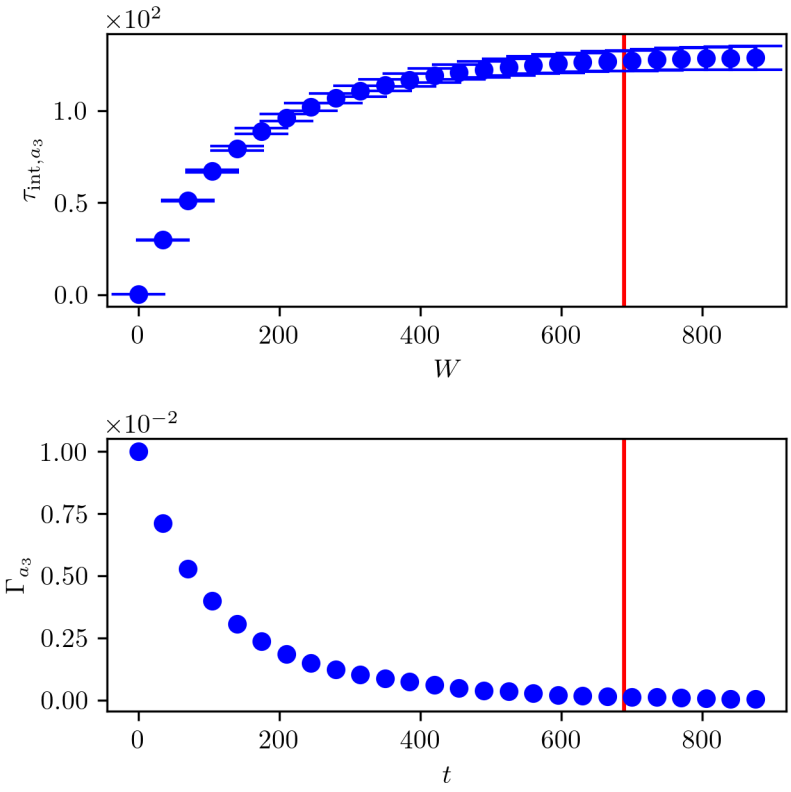


Figure B.21: Provided by [28], the IACT τ_{int,a_3} at summation windows W and the estimated autocorrelation function Γ_{a_3} at lag t of samples $a_3 \sim \pi(\cdot|\boldsymbol{y})$ from the t-walk for the approximated forward model.

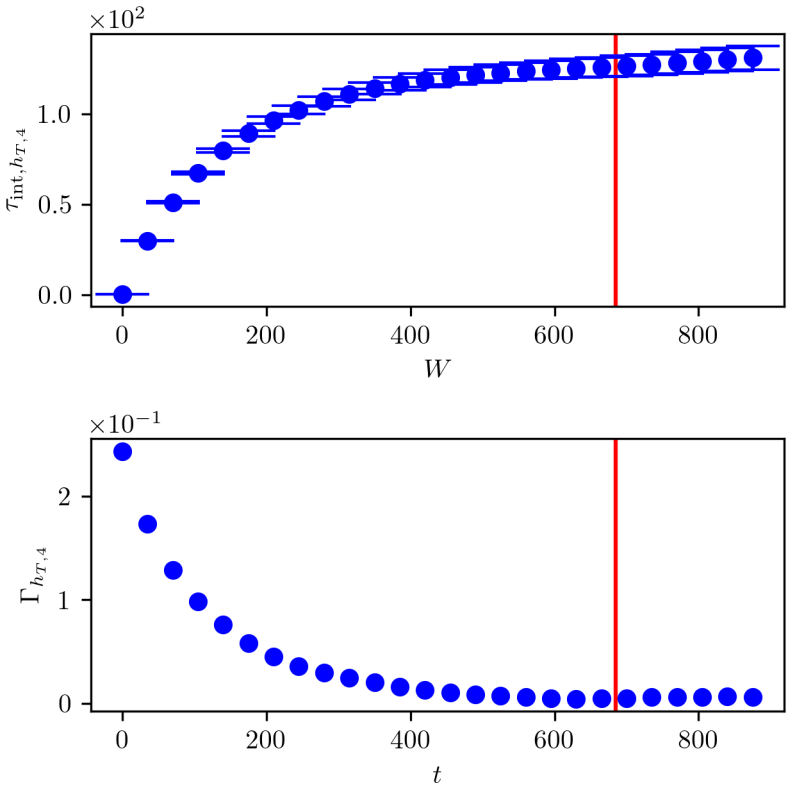


Figure B.22: Provided by [28], the IACT $\tau_{\text{int},h_{T,4}}$ at summation windows W and the estimated autocorrelation function $\Gamma_{h_{T,4}}$ at lag t of samples $h_{T,4} \sim \pi(\cdot|\boldsymbol{y})$ from the t-walk for the approximated forward model.

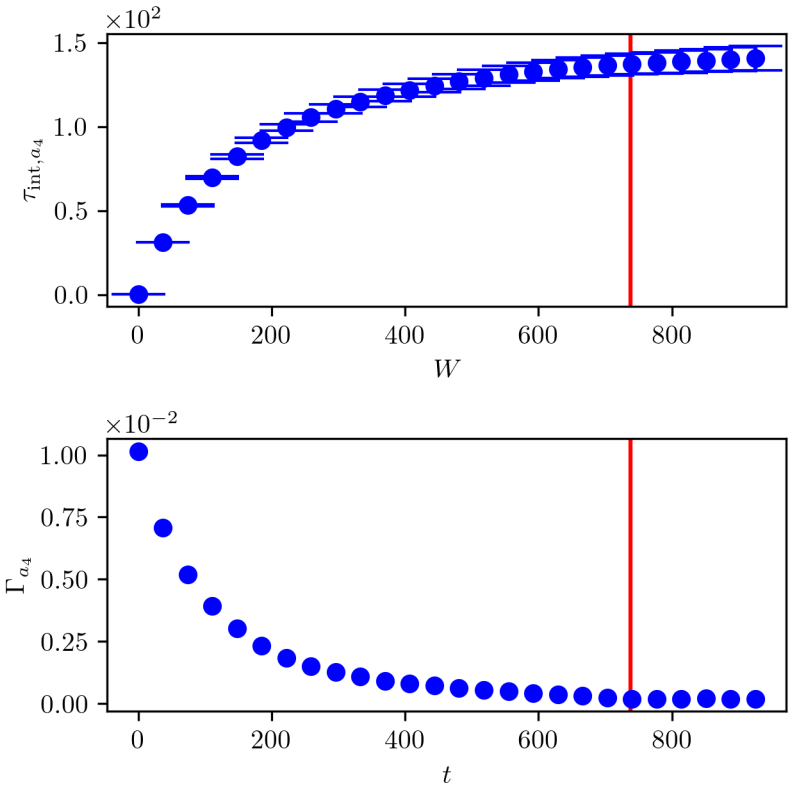


Figure B.23: Provided by [28], the IACT τ_{int,a_4} at summation windows W and the estimated autocorrelation function Γ_{a_4} at lag t of samples $a_4 \sim \pi(\cdot|\boldsymbol{y})$ from the t-walk for the approximated forward model.

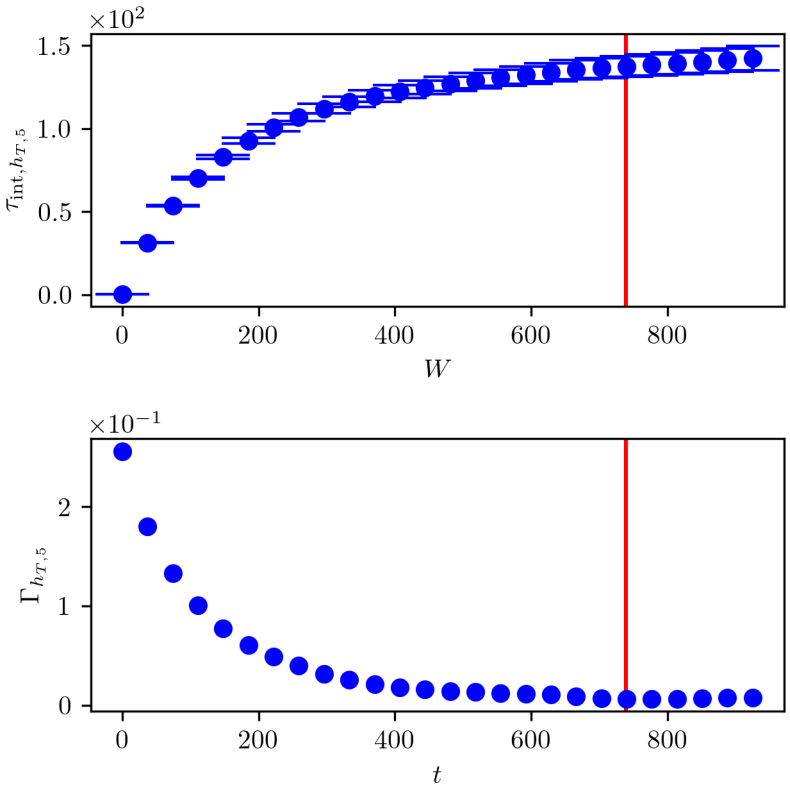


Figure B.24: Provided by [28], the IACT $\tau_{\text{int},h_{T,5}}$ at summation windows W and the estimated autocorrelation function $\Gamma_{h_{T,5}}$ at lag t of samples $h_{T,5} \sim \pi(\cdot|\boldsymbol{y})$ from the t-walk for the approximated forward model.

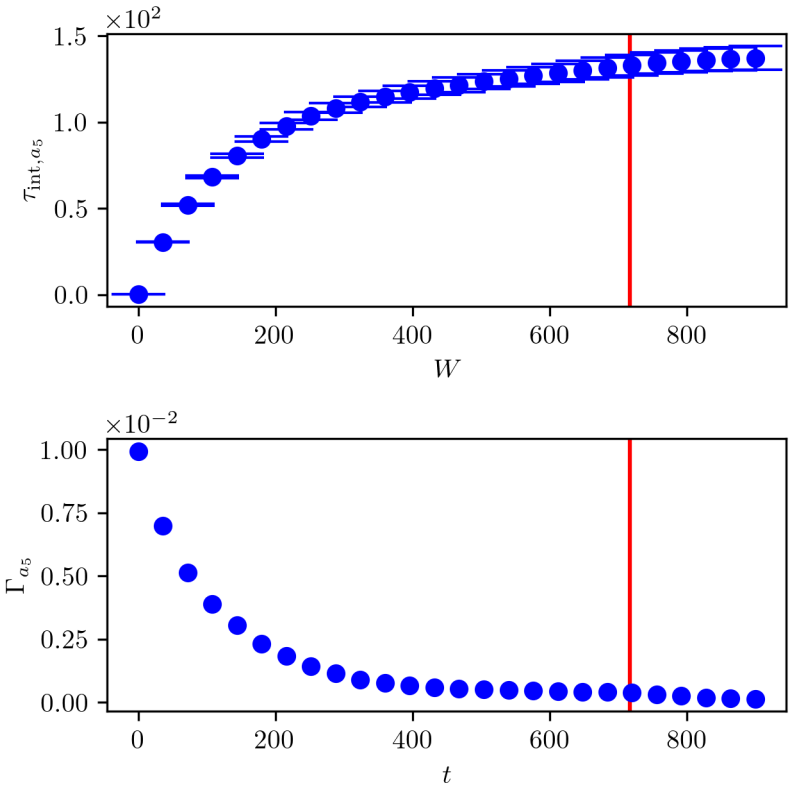


Figure B.25: Provided by [28], the IACT τ_{int,a_5} at summation windows W and the estimated autocorrelation function Γ_{a_5} at lag t of samples $a_5 \sim \pi(\cdot|\boldsymbol{y})$ from the t-walk for the approximated forward model.

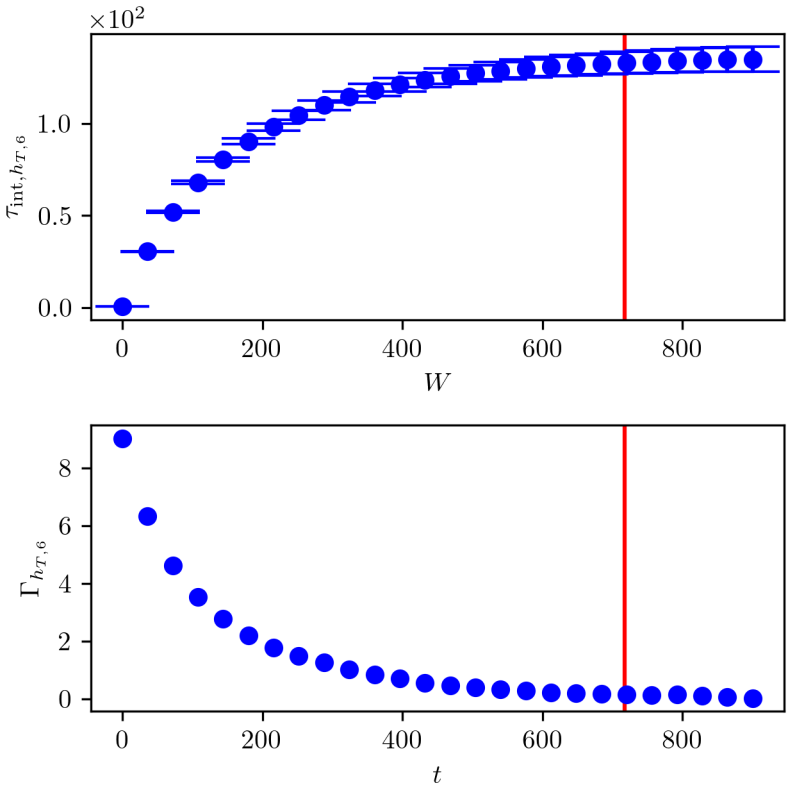


Figure B.26: Provided by [28], the IACT $\tau_{\mathrm{int},h_{T,6}}$ at summation windows W and the estimated autocorrelation function $\Gamma_{h_{T,6}}$ at lag t of samples $h_{T,6} \sim \pi(\cdot|\boldsymbol{y})$ from the t-walk for the approximated forward model.

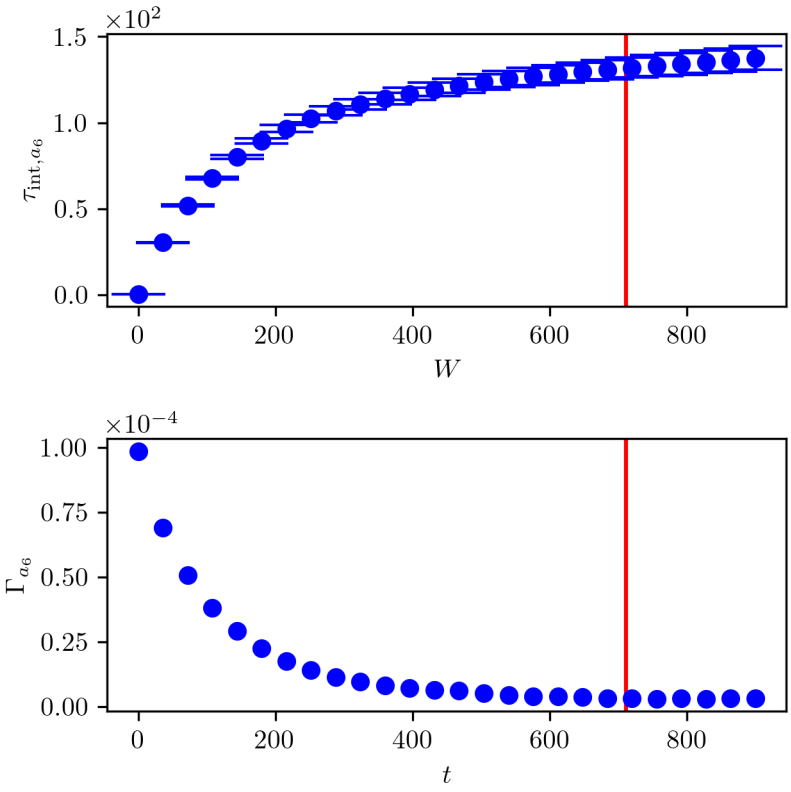


Figure B.27: Provided by [28], the IACT τ_{int,a_6} at summation windows W and the estimated autocorrelation function Γ_{a_6} at lag t of samples $a_6 \sim \pi(\cdot|\boldsymbol{y})$ from the t-walk for the approximated forward model.