GEORG-AUGUST-UNIVERSITÄT
GÖTTINGEN

MASTER'S THESIS

submitted in partial fulfillment of the
requirements for the study program

"Applied Data Science M. Sc."

Stochastic Variational Inference for Structured Additive Distributional Regression in Peak-over-Threshold Extreme Value Modeling

---

MARCEL ALEXANDER GAWEDA

INSTITUTE OF COMPUTER SCIENCE

MASTER'S THESES
of the Chair of Statistics
at the Georg-August-Universität Göttingen

Georg-August-Universität Göttingen
Institute of Computer Science

Goldschmidtstraße 7
37077 Göttingen
Germany

☎ +49(551) 39-172000
FAX +49(551)39-14403
✉ office@cs.uni-goettingen.de
www.informatik.uni-goettingen.de

First Supervisor: Prof. Dr. Thomas Kneib
Second Supervisor: Dr. Johannes Söding

Abstract

This thesis investigates the applicability of Stochastic Variational Inference (SVI) for Bayesian Structured Additive Distributional Regression (SADR) with Generalized Pareto (GP) distributed responses, comparing it against Markov Chain Monte Carlo (MCMC). The parameter-dependent support of the GP distribution poses a computational challenge for SVI, as responses falling outside the support result in undefined gradients that impede optimization. A modified GP distribution is proposed to handle these out-of-support (OOS) cases by returning finite values that implicitly penalize parameter values causing OOS responses. The comparative analysis uses both a simulation study with varying sample sizes ($N = 250, N = 500, N = 1000$) *and a case study using the Danish Fire Insurance dataset. For the simulation study, the Wasserstein distance (WD) and Sinkhorn distance (SD) quantify the dissimilarity between posterior distributions inferred by SVI and MCMC. The case study implements a two-stage Peaks-over-Threshold (PoT) approach, fitting an Asymmetric Laplace distribution to determine thresholds and then applying the GP distribution to model exceedances. Results indicate that SVI with a full-covariance multivariate normal variational distribution adequately approximates the joint posterior distribution and the GP shape parameter, while exhibiting persistent discrepancies for the GP scale parameter and systematically underestimating posterior uncertainty. SVI demonstrates substantial computational efficiency compared to MCMC. Results indicate that SVI with the modified GP distribution constitutes a viable approach for Bayesian SADR with extreme value data when computational efficiency takes precedence over accuracy and precise uncertainty quantification. Further research into more sophisticated variational distributions for this use-case is warranted to achieve both computational efficiency and enhanced estimation precision.*

Contents

List of Figures

List of Tables

List of Algorithms

List of Abbreviations

AL	Asymmetric Laplace	5
ALD	Asymmetric Laplace distribution	5
CAVI	Coordinate Ascend Variational Inference	26
DKK	Danish Krone	46
DMN	Degenerate Multivariate Normal	17
ELBO	Evidence Lower Bound	4
ESS	Effective Sample Size	6
FCMN	Full Covariance Multivariate Normal	4
GAMLSS	Generalized Additive Models for Location, Scale, and Shape	2
GEV	Generalized Extreme Value distribution	5
GP	Generalized Pareto	3
GPD	Generalized Pareto distribution	3
GPU	Graphical Processing Unit	6
HDI	Highest Density Interval	32
IG	Inverse Gamma	18
KL	Kullback-Leibler	4
kwargs	keyword arguments	35
lr	learning rate	35
MC	Monte Carlo	27
MCMC	Markov Chain Monte Carlo	4
MFVI	Mean-Field Variational Inference	25
MN	Multivariate Normal	31
nan	not a number	36
NUTS	No-U-Turn Sampler	36
OOS	Out-of-Support	5
PDF	Probability Density Function	5
PoT	Peaks over Threshold	4
P-splines	Penalized B-splines	3
QQ	Quantile-Quantile	6

Chapter 1

Introduction

BAYESIAN linear regression provides a probabilistic framework for inferring the relationship between responses and covariates. Consider the random vector

$$\vec{Y} = (Y_i)_{i=1}^{N}, \tag{1.1}$$

where the N random variables are independent, but not necessarily identical, continuous random variables. For each i, let $y_i \in \mathbb{R}$ denote the realization of Y_i, i. e. the observed response, and define $\vec{y} = (y_1, \ldots, y_N)^T \in \mathbb{R}^N$ as the vector of all realizations. The vector representing a univariate covariate with N measurements is represented as $\vec{x} = (x_1, \ldots, x_N)^T \in \mathbb{R}^N$, where x_i denotes an individual measurement. For observation i, the vector $\vec{\mathfrak{x}}_i = (x_{i1}, \ldots, x_{iP})^T \in \mathbb{R}^P$ represents P multivariate covariate measurements, where $j = 1, \ldots, P$. The design matrix $\mathbf{X} \in \mathbb{R}^{N \times \tilde{P}}$ consists of $N \times P$ covariate measurements plus an intercept column, resulting in $\tilde{P} = P + 1$ features. Standard regression analysis typically models the conditional mean of responses given the design matrix $\mathbf{X}$ and the regression coefficients $\vec{\beta}$ with $\vec{\beta} \in \mathbb{R}^{\tilde{P}}$(Kneib et al., 2023). In Bayesian regression $\vec{\beta}$ is assumed to be a random variable with prior distribution $p(\vec{\beta})$ and the responses are assumed to follow the response distribution $p(\vec{y}|\mathbf{X}, \vec{\beta})$. The joint posterior distribution over the $\tilde{P}$ parameters is obtained through Bayes' theorem as

$$p(\vec{\beta}\,|\,\vec{y}, \mathbf{X}) = \frac{p(\vec{y}|\mathbf{X}, \vec{\beta})\, p(\vec{\beta})}{\int p(\vec{y}|\mathbf{X}, \vec{\beta})\, p(\vec{\beta})\, d\vec{\beta}},$$

(Gelman et al., 2014, pp. 353–354). The standard regression's limitation of modeling only the conditional mean motivated the development of more flexible frameworks that capture the entire response distribution, e. g. distributional regression (Kneib et al., 2023).

Distributional regression goes beyond modeling the conditional mean by characterizing the full conditional density through various modeling approaches. These approaches differ in how they characterize a distribution—from parametric to non-parametric distributional properties. To name a few, the main methodological frameworks include Generalized Additive Models for Location, Scale, and

Shape (GAMLSS), conditional transformation models, density regression, and quantile/expectile regression. GAMLSS provides a parametric framework that models all the parameters of the distribution (Kneib et al., 2023). To obtain an estimate of the full conditional distribution over the responses given the covariates, this thesis uses GAMLSS.

Briefly speaking, GAMLSS does distributional regression for a K-parametric response distribution

$$\overrightarrow{y} \overset{\text{ind.}}{\sim} p(\overrightarrow{y}|\overrightarrow{\theta_1},\ldots,\overrightarrow{\theta_K}),$$

by linking a linear predictor $\overrightarrow{\eta}_k = \mathbf{X}_k \overrightarrow{\beta}_k$ to the k-th distribution's parameter through the inverse of a parameter-specific strictly monotonic link function $g_k(\cdot)$, i. e. $\overrightarrow{\theta}_k = g_k^{-1}(\overrightarrow{\eta}_k)$, with $k = 1,\ldots,K$, $\overrightarrow{\theta_k} \in \mathbb{R}^N$, $\overrightarrow{\eta}_k \in \mathbb{R}^N$, $\mathbf{X}_k \in \mathbb{R}^{N\times\widetilde{P}_k}$ where $\widetilde{P}_k$ denotes the number of features used to model the distribution's k-th parameter (Stasinopoulos et al., 2024, p. 23).[1] In order to capture non-linear relations between the features and the response a Penalized B-splines (P-splines) basis transformation can be employed (Lang & Brezger, 2004). If used, the simple linear predictor is then exchanged with a Structured Additive Predictors (SAP). To model the full conditional response distribution, this study uses a GAMLSS regression model with structured additive predictors using P-splines within a Bayesian framework (Stasinopoulos et al., 2024, p. 53). In the following, the Bayesian GAMLSS regression model with P-spline SAP will be referred to as Structured Additive Distributional Regression (SADR) as done in Callegher et al. (2025). The details of SADR are presented in Chapter 4. Within SADR, this thesis assumes that the responses follow an extreme value distribution—the Generalized Pareto distribution (GPD).

To put it briefly and, for now, considered in isolation from the SADR framework, let $Y_1, Y_2, \ldots, Y_N$ be a sequence of independent and identical random variables. Furthermore, assume that each Y_i with $i = 1,\ldots,N$ from the sequence $Y_1, Y_2, \ldots, Y_N$ follows a common distribution function F, and suppose that F is in the domain of attraction of an extreme value distribution. Then for a threshold u sufficiently close to the right endpoint y_F of the support of F, i. e. for a sufficiently high threshold, the conditional threshold exceedance distribution function of $Y_i - u$, given $Y_i > u$, converges to a Generalized Pareto (GP) distribution:

$$\lim_{u\to x_F} P(Y_i - u|Y_i > u) = H(y_i;\sigma,\xi),$$

with $u \in \mathbb{R}, \sigma \in \mathbb{R}_+, \xi \in \mathbb{R}$, and $H(y_i)$ being the GPD's cumulative distribution function:

$$H(y_i) = \begin{cases} 1 - \left(1 + \xi\frac{y_i}{\sigma}\right)^{-1/\xi} & \text{if } \xi \neq 0, \\ 1 - \exp\left(-\frac{y_i}{\sigma}\right) & \text{if } \xi = 0, \end{cases}$$

(Balkema and de Haan, 1974; Coles, 2001, pp. 75–76; Pickands III, 1975). The support $\mathscr{X}$ of the GPD

[1] The notation $\overrightarrow{y} \overset{\text{ind.}}{\sim} p(\overrightarrow{y} \mid \overrightarrow{\theta}_1,\ldots,\overrightarrow{\theta}_K)$ should not be interpreted as a multivariate distribution. Rather, it denotes that each element y_i in $\overrightarrow{y}$ is independently distributed according to the distribution p with the corresponding elements from the parameter vectors $\overrightarrow{\theta}_1,\ldots,\overrightarrow{\theta}_K$.

depends on its parameters:

$$\mathscr{X} = \begin{cases} \{y_i \in \mathbb{R} : y_i \geq 0\} & \text{if } \xi \geq 0, \quad (1.2) \\ \{y_i \in \mathbb{R} : 0 \leq y_i \leq -\sigma/\xi\} & \text{if } \xi < 0\,, \quad (1.3) \end{cases}$$

(Embrechts et al., 1997, p. 162–164).

This GPD parameterization uses the two-parameter scale-shape notation, $\text{GP}(y_i|\sigma,\xi)$, instead of the three-parameter location-shape notation, $\text{GP}(y_i|\mu,\sigma,\xi)$, as in Coles (2001, pp. 75–76), because when Y_i is centered around the threshold u, the location parameter of the GPD becomes zero, since the location parameter corresponds to the threshold. The statistical procedure to define a threshold and model the exceedances/peaks over the threshold with a $\text{GP}(\sigma,\xi)$ is known as Peaks over Threshold (PoT) (Balkema and de Haan, 1974; Bourguignon et al., 2015, p. 42; Pickands III, 1975). A more comprehensive theoretical introduction to extreme value modeling is presented in Chapter 2.

This study's model design yields a non-conjugate Bayesian problem for joint posterior estimation. The non-conjugate Bayesian GAMLSS problem typically approaches the posterior inference by using Markov Chain Monte Carlo (MCMC) methods (Stasinopoulos et al., 2024, p. 132; Stasinopoulos et al., 2024, p. 143). Instead, in order to infer the posterior, this study uses a variant of Stochastic Variational Inference (SVI) that makes use of automatic differentiation and the reparameterization trick (Kingma & Welling, 2013; Kucukelbir et al., 2016). Other variants of SVI make use of the score function estimator (a. k. a. REINFORCE) (Murphy, 2023, p. 268), Stein variational gradient descent (Liu & Wang, 2016), or the Stick-the-Landing gradient estimator (Roeder et al., 2017). An advantage of using the reparameterization trick for SVI rather than other variants is that it is natively supported in TensorFlow Probability's (Dillon et al., 2017) probability distribution classes, allowing for a straightforward implementation in Python.

As explained by Blei et al., unlike MCMC, which simulates the posterior by drawing samples from a Markov chain that asymptotically converges to the true posterior distribution, SVI reformulates posterior inference as an optimization problem, finding the closest approximate distribution to the true posterior. SVI uses Stochastic Gradient Descent (SGD) to minimize a loss function—the negative Evidence Lower Bound (ELBO)—which implicitly minimizes the Kullback-Leibler (KL) divergence between the approximate distribution, called the variational distribution, and the possibly unknown true posterior. The key advantage of SVI lies in its computational speed, allowing for faster convergence than MCMC methods. However, MCMC asymptotically converges against the exact samples of the true posterior as the number of iterations increases, while SVI's posterior approximation is limited by the expressiveness of the chosen variational distribution, which, when not expressive enough, cannot converge against the true posterior, even with infinite computational resources. The choice between these methods therefore represents a trade-off between computational speed and posterior accuracy (Blei et al., 2017). The variational distribution of choice for this thesis is a Full Covariance Multivariate Normal (FCMN) distribution. Chapter 5 establishes the groundwork for SVI by exploring Variational Inference (VI) generally, motivating SVI, and explaining the SVI algorithm in detail.

Estimating the scale and shape parameters of the GPD may be challenging because these parameters determine the distribution's support. In cases where the shape parameter is negative, the support has an upper bound (see (1.3)) that depends on both the scale and the shape. Consequently, an estimate might yield an upper bound that is smaller than the largest observation, thus violating the support constraint and producing an Out-of-Support (OOS) non-sensical result (Ashkar & Nwentsa Tatsambon, 2007; Castillo & Hadi, 1997; de Zea Bermudez & Kotz, 2010a; Dupuis & Tsao, 1998). de Zea Bermudez and Kotz (2010a) list modifications applied to non-Bayesian methods that address the problem. Bayesian approaches that specifically address this problem do not appear to have been developed (cf. de Zea Bermudez and Kotz, 2010b). Studies dealing specifically with SAP in the context of PoT extreme value inference using the GPD in a non-Bayesian setting (cf. Konzen et al., 2021; Youngman, 2020) and Bayesian setting (cf. Das et al., 2010; Randell et al., 2016; Zanini et al., 2020) do not address this issue and infer the GPD's parameters naïvely regarding potential Out-of-Support (OOS) cases.

However, SVI faces a fundamental challenge when the model parameters determine the support of the response distribution. During optimization, parameter estimates may inadvertently place observed responses outside the distribution's support. This creates a critical problem: the GPD's Probability Density Function (PDF) is defined as 0 for OOS responses, and consequently $\log(\text{PDF}(0)) = \log(0) = -\infty$. The resulting infinite loss prevents proper gradient-based optimization because the gradients become undefined. Currently, there is no solution in the literature that addresses this issue for posterior estimation using SVI. A simple approach might be to skip parameter estimates that lead to OOS observations. However, this results in an inefficient computational behavior, undermining the computational speed advantage of SVI (G. Callegher, personal communication, September 10, 2024). Instead, this work proposes to penalize the occurrence of OOS cases, but not by modifying the loss function, but by modifying the GP's log PDF to handle OOS cases by returning a value other than $-\infty$ and one that results in an increase in the loss function, and thus an implicit penalty for parameter values that cause OOS responses. This approach, first implemented for the Generalized Extreme Value distribution (GEV) by J. Brachem (personal communication, April 01, 2025), which also has parameter-dependent support, and adapted for the GPD by G. Callegher (personal communication, January 06, 2025), preserves algorithmic modularity by implementing changes only at the source of the problem—the GPD—while retaining the original SVI optimization framework. Details about the modified GPD are shown in Section 6.6.

To identify and fit the extreme values of a dataset to the GPD—to perform PoT inference—this thesis adapts the two-stage maximum likelihood inference approach of Youngman (2019) to Bayesian inference. The first step is to determine a threshold u to identify excesses. This is done in Youngman (2019) by performing quantile regression by fitting the Asymmetric Laplace distribution (ALD) to the observed data at a user-specified quantile level τ. Although the aforementioned Bayesian SADR regression framework and SVI posterior estimation were introduced for the GPD, SADR is a modular approach so that the Asymmetric Laplace (AL) distribution can be fitted by changing the response distribution accordingly, i. e. the responses are assumed to follow the ALD and the ALD's parameters are linked to a SAP. This implies that the threshold u is covariate-dependent. Rather than employing a single fixed value, the threshold is

modeled as a continuous P-spline function and evaluated at each observation point. After determining the covariate-dependent threshold and identifying the responses that exceed the threshold, the second stage in Youngman (2019) fits the GPD. Bayesian quantile regression with ALD and a detailed description of the two-stage approach are discussed in Chapter 3, and selecting a sufficiently high threshold is discussed in Section 6.9.4.

The core of this thesis is the comparative analysis of the posterior distribution inference methods for SAP parameters in the SADR framework with GP distributed responses. Contrasting SVI using the modified GPD and a FCMN variational distribution against a MCMC approach using the unmodified naïve GPD. A comparison between SVI and MCMC methods for SADR has been done in Callegher et al. (2025), but for binary, Gamma, and Negative Binomial distributed responses. This thesis follows the comparative approach in Callegher et al. (2025) and performs a simulation study and a case study with real data.

The simulation study systematically evaluates the performance difference between SVI and MCMC through iterative testing on simulated GP-distributed univariate responses linked to a one-dimensional covariate. Data generation is conducted repeatedly across varying sample sizes. The Wasserstein distance (WD)(Peyré & Cuturi, 2020, p. 23) serves as a performance metric, quantifying the dissimilarity between posterior distribution samples derived from both estimation methods. Due to computational resource constraints, different from Callegher et al. (2025), this study utilizes the Sinkhorn distance (SD) (Cuturi, 2013) as an approximation of the WD for handling multidimensional joint distributions and posterior marginals. Furthermore, due to computational constraints, the MCMC samples undergo thinning because the sample size required for an adequate Effective Sample Size (ESS) demands computational resources unavailable for this thesis when calculating the SD for multidimensional posteriors.

The case study uses the *Danish Fire Insurance* dataset (obtained through the R package EVIR (Pfaff et al., 2018)) and implements the two-stage PoT approach. The GPD is fitted to the same exceedances for both SVI and MCMC, enabling direct comparative analysis. Posterior estimation results are evaluated through visualizations of the marginal posteriors' estimated kernel densities and the estimated covariate-dependent GPD scale and shape parameters, while the goodness of fit is evaluated with a GPD-specific Quantile-Quantile (QQ) plot.

The SVI algorithm is implemented in PYTHON using JAX (Bradbury et al., 2018) to provide an efficient Graphical Processing Unit (GPU)-accelerated SVI algorithm.

Chapter 6 presents detailed implementation specifications for the simulation and case study; the MCMC and SVI configuration parameters; the SADR model configurations; the WD and SD dataset characteristics; the posterior sample thinning procedure; the GPD-specific QQ plot construction; and additional implementation details relevant to this thesis.

Chapter 2

Extreme Value Modeling

Extreme value theory provides a mathematical framework for characterizing the limiting behavior of probability distributions (Bourguignon et al., 2015, p. 42). This chapter presents the foundational theory for modeling extreme values, focusing on the derivation of the GEV distribution and the GPD.

2.1 Generalized Extreme Value Distribution

As presented by Coles (2001), for a sequence of N independent and identical random variables $(Y_i)_{i=1}^N$ with common distribution function F, consider the maximum

$$M_N = \max\{Y_1, Y_2, \ldots, Y_N\}. \tag{2.1}$$

The distribution of M_N is given by

$$P(M_N \leq y_i) = P(Y_1 \leq y_i, Y_2 \leq y_i, \ldots, Y_N \leq y_i) = [F(y_i)]^N. \tag{2.2}$$

For most underlying distributions F, the limiting distribution of M_N as $N \to \infty$ is degenerate. However, appropriately normalized maxima lead to non-degenerate limiting distributions. This approach is commonly referred to as the block maxima method in extreme value analysis (pp. 45–49).

The fundamental result in extreme value theory, established by Fisher and Tippett (Fisher & Tippett, 1928; Gnedenko, 1943) and elaborated by Bourguignon et al. (2015), states that if there exist sequences of normalizing constants $a_n > 0$ and b_n such that

$$P\left(\frac{M_N - b_N}{a_N} \leq y_i\right) = [F(a_N y_i + b_N)]^N \to G(y_i), \tag{2.3}$$

as $N \to \infty$ for some non-degenerate distribution function G, then G belongs to one of three types of

possible limiting distributions for maxima, i. e. extreme value distributions:

1. Gumbel (Type I):

$$\Lambda(y_i) = \exp(-\exp(-y_i)), \quad \text{if } -\infty < y_i < \infty, \tag{2.4}$$

2. Fréchet (Type II):

$$\Phi_\alpha(y_i) = \begin{cases} 0, & \text{if } y_i \leq 0, \\ \exp(-y^{-\alpha}), & \text{if } y_i > 0, \end{cases} \tag{2.5}$$

   where $\alpha > 0$,

3. Weibull (Type III):

$$\Psi_\alpha(y_i) = \begin{cases} \exp(-(-y_i)^\alpha), & \text{if } y_i \leq 0, \\ 1, & \text{if } y_i > 0, \end{cases} \tag{2.6}$$

   where $\alpha > 0$ (pp. 41–42).

If F meets these criteria, it follows that F is in the domain of attraction for G, expressed as $F \in DA(G)$ (Bourguignon et al., 2015, p. 41).

The three extreme value distribution types can be combined into a single parametric family, known as the Generalized Extreme Value (GEV) distribution. This generalized representation has the distribution function

$$G(y_i) = \exp\left\{-\left[1 + \xi\left(\frac{y_i - \mu}{\sigma}\right)\right]^{-1/\xi}\right\}, \tag{2.7}$$

defined for

$$\mathscr{X} = \begin{cases} y_i \in [\mu - \frac{\sigma}{\xi}, +\infty), & \text{if } \xi > 0, \\ y_i \in (-\infty, \infty), & \text{if } \xi = 0, \\ y_i \in (-\infty, \mu - \frac{\sigma}{\xi}], & \text{if } \xi < 0, \end{cases} \tag{2.8}$$

where $\mu \in \mathbb{R}$, $\sigma \in \mathbb{R}_+$, and $\xi \in \mathbb{R}$. The parameters μ, σ, and ξ are the location, scale, and shape parameters, respectively. The shape parameter ξ determines the type of the distribution: $\xi = 0$ corresponds to the Gumbel distribution (Type I), $\xi > 0$ corresponds to the Fréchet distribution (Type II), and $\xi < 0$ corresponds to the Weibull distribution (Type III). The case $\xi = 0$ is interpreted in the limit as $\xi \to 0$, giving

$$G(y_i) = \exp\left\{-\exp\left[-\left(\frac{y_i - \mu}{\sigma}\right)\right]\right\} \quad \text{for } -\infty < y_i < \infty, \tag{2.9}$$

(Coles, 2001, pp. 47–49).

2.2 Generalized Pareto Distribution

While the GEV distribution models block maxima, the GPD arises from modeling exceedances over a sufficiently high threshold. This approach, known as PoT, enables a more effective use of data than only the block maxima, while the block maxima strategy is more robust to dependencies between observations (Bourguignon et al., 2015, p. 42). The GPD is mathematically justified by the Pickands-Balkema-de Haan theorem (Bourguignon et al., 2015, p. 42) which is attributed to Balkema and de Haan (1974) and Pickands III (1975).

As presented by Bourguignon et al. (2015), the Pickands-Balkema-de Haan theorem states that given a sequence of N independent and identical random variables $(Y_i)_{i=1}^N$ with distribution function F, and threshold u, then, if F is in the domain of attraction of an extreme value distribution G and if u approaches the upper endpoint of the support of F, the conditional distribution of excesses $Y_i - u$, given that $Y_i > u$, converges against a GPD:

$$F \in DA(G) \text{ iff } \lim_{u \to y_{iF}} \sup_{0 \le y_i < y_{iF} - u} \left| F_{Y_i - u | Y_i > u}(y_i) - H_{y_i;\sigma,\xi}(x) \right| = 0, \tag{2.10}$$

where y_{iF} is the right endpoint of the support of F, and

$$H(y_i;\sigma,\xi) = \begin{cases} 1 - \left(1 + \xi \frac{y_i}{\sigma}\right)^{-1/\xi}, & \text{if } \xi \neq 0, \\ 1 - \exp\left(-\frac{y_i}{\sigma}\right), & \text{if } \xi = 0, \end{cases} \tag{2.11}$$

being the GPD cumulative distribution function with scale parameter $\sigma > 0$ and shape parameter $\xi \in \mathbb{R}$ (pp. 42–43).

The support of the distribution depends on the shape parameter ξ:

$$\mathscr{X} = \begin{cases} \{y_i \in \mathbb{R} : y_i \ge 0\} & \text{if } \xi \ge 0, \\ \{y_i \in \mathbb{R} : 0 \le y_i \le -\sigma/\xi\} & \text{if } \xi < 0, \end{cases} \tag{2.12}$$

(Embrechts et al., 1997, p. 162–164).

Similarly to the GEV distribution, Bourguignon et al. (2015, p. 42) show that the GPD can be separated into three distinct types depending on the value of the shape parameter ξ, if interpreting $\xi = 0$ as the limit $\xi \to 0$,:

1. Exponential (Type I):

$$H(y_i) = \begin{cases} 0, & \text{if } y_i < 0, \\ 1 - \exp(-y_i), & \text{if } y_i \ge 0, \end{cases} \tag{2.13}$$

if $\xi = 0$,

2. Pareto (Type II):

$$H(y_i) = \begin{cases} 0, & \text{if } y_i \leq 0, \\ 1 - y_i^{-1/\xi}, & \text{if } y_i > 0, \end{cases} \tag{2.14}$$

where $\xi > 0$,

3. Beta (Type III):

$$H(y_i) = \begin{cases} 1 - (-y_i)^{-1/\xi}, & \text{if } y_i < 0, \\ 1, & \text{if } y_i \geq 0, \end{cases} \tag{2.15}$$

where $\xi < 0$.

The shape parameter ξ indicates the heaviness of the tail, with the tail becoming heavier for larger values of ξ (Bourguignon et al., 2015, p. 43).

Chapter 3

Youngman's Two-Stage Approach

Youngman (2019) proposes a two-stage procedure for modeling exceedances over thresholds. The procedure first infers the threshold using quantile regression and then models excesses above this threshold using the GPD. While Youngman implements it in an SADR manner for likelihood-inference, this thesis embeds it in the Bayesian SADR approach detailed in Chapter 4.

First stage: The quantile regression is based on the approach presented in Yu and Moyeed (2001), where Bayesian quantile regression is performed using the ALD as the response distribution. Let Y be a random variable, and y its realization, then the ALD is parameterized as $\text{AL}(y|u,\psi,\tau)$ with u,ψ,τ being the location, scale, and the regression quantile-level, respectively, and $Y \sim \text{AL}(u,\psi,\tau)$. Its PDF is defined as

$$f(y;\mu,\sigma,p) = \frac{\tau(1-\tau)}{\psi} \exp\left\{-\rho_\tau\left(\frac{y-\mu}{\psi}\right)\right\}, \tag{3.1}$$

with ρ being a check function depended on τ. For a user-specified quantile level τ, a univariate covariate x, and a linear predictor $\eta(x)$ for the location parameter μ, inferring the ALD's location parameter for the response y given the covariate x is equivalent to inferring the conditional quantile $\delta_\tau(y|x)$ at the quantile level τ (Youngman, 2019). In accordance with the notation in Youngman (2019), the ALD is denoted as $\text{AL}(y|u_\tau,\psi)$ where the regression quantile-level τ is shifted into the subscript of the location to symbolize the direct influence of τ on the location and also on the inferred conditional quantile. Regarding the check function ρ, Youngman (2019) uses a modified check function

$$\rho_{\tau,\mathfrak{c}}(u) = \begin{cases} (\tau-1)(2u+\mathfrak{c}), & \text{for } u < \mathfrak{c} \text{ [sic!]}, \\ 0.5(1-\tau)u^2/\mathfrak{c}, & \text{for } -\mathfrak{c} \le u < 0, \\ 0.5\tau u^2/\mathfrak{c}, & \text{for } 0 \le u < \mathfrak{c}, \\ \tau(u-0.5\mathfrak{c}), & \text{for } \mathfrak{c} \le u, \end{cases} \tag{3.2}$$

where $\mathfrak{c} > 0$ is a small scalar controlling the amount of smoothing at the origin. The modified check function is meant to improve numerical stability in optimization (Oh et al., 2011), however, the implementation in this thesis uses the unmodified check function (Yu & Zhang, 2005),

$$\rho_\tau(u) = \begin{cases} (\tau - 1)u, & \text{if } u < 0, \\ \tau u, & \text{if } u \geq 0, \end{cases} \tag{3.3}$$

as no numerical instabilities have been observed when using the unmodified check function.

Second stage: Given N inferred covariate-dependent thresholds $u_\tau(x_i)$ for $i, \ldots, N$, the respective responses that exceed the threshold (the exceedances over the threshold) are determined and assumed to follow a GPD (see Chapter 2).

A technical embedding of the two-stage approach within the Bayesian SADR model is shown in Chapter 4.

Chapter 4

Bayesian Structured Additive Distributional Regression

4.1 Generalized Additive Models for Location, Scale, and Shape

GAMLSS provides a flexible framework for modeling not only the conditional mean of a response variable, but all parameters of its conditional distribution (Rigby & Stasinopoulos, 2005). This approach extends traditional generalized linear models and generalized additive models, which focus exclusively on modeling the conditional expectation (Stasinopoulos et al., 2024, pp. 17–19).

Consider the random vector $\overrightarrow{Y} = (Y_i)_{i=1}^N$ where all N random variables are independent, but not necessarily identical, continuous random variables. For each i, let y_i denote the realization of Y_i, i. e. the observed response, and define $\overrightarrow{y} = (y_1, \ldots, y_N)^T \in \mathbb{R}^N$ as the vector of all realizations. Furthermore, let $\overrightarrow{x} = (x_1, \ldots, x_N)^T \in \mathbb{R}^N$ denote N univariate covariate measurements, where x_i denotes an individual observation of the covariate and let $\overrightarrow{\mathfrak{x}}_i = (x_{i1}, \ldots, x_{iP})^T \in \mathbb{R}^P$ be the vector for multivariate covariate measurements for an observation i. Then, the design matrix $\mathbf{X} \in \mathbb{R}^{N \times \widetilde{P}}$ consists of $N \times P$ covariate measurements plus an intercept column, resulting in $\widetilde{P} = P + 1$ features.

GAMLSS assumes that the conditional distribution of the response variable follows a parametric distribution with $\mathbf{K}$ parameters:

$$\overrightarrow{y} \overset{\text{ind.}}{\sim} p(\overrightarrow{y} | \overrightarrow{\theta_1}, \ldots, \overrightarrow{\theta_K}), \tag{4.1}$$

where, in this thesis, $p(\cdot)$ denotes a continuous probability measure, but can represent discrete, or mixed discrete-continuous distributions too (Stasinopoulos et al., 2024, p. 19).[1] Each parameter vector $\overrightarrow{\theta}_k \in \mathbb{R}^N$ is related to a predictor vector $\overrightarrow{\eta}_k \in \mathbb{R}^N$ through a parameter-specific, strictly monotonic link function

[1] The notation $\overrightarrow{y} \overset{\text{ind.}}{\sim} p(\overrightarrow{y} \mid \overrightarrow{\theta}_1, \ldots, \overrightarrow{\theta}_K)$ should not be interpreted as a multivariate distribution. Rather, it denotes that each element y_i in $\overrightarrow{y}$ is independently distributed according to the distribution p with the corresponding elements from the parameter vectors $\overrightarrow{\theta}_1, \ldots, \overrightarrow{\theta}_K$.

$g_k(\cdot)$:

$$\vec{\eta}_k = g_k(\vec{\theta}_k), \quad \text{or equivalently,} \quad \vec{\theta}_k = g_k^{-1}(\vec{\eta}_k), \tag{4.2}$$

for $k = 1,\dots,K$ with $g_k^{-1}(\cdot)$ being the response function. The link functions are chosen such that the parameter space constraints are satisfied. For example, if $\vec{\theta}_k$ represents the scale parameters, a logarithmic link function $g_k(\vec{\theta}_k) = \log(\vec{\theta}_k)$ or equivalently $\vec{\theta}_k = \exp(\vec{\eta}_k)$ would be appropriate (Stasinopoulos et al., 2024, pp. 20–21).

In the standard GAMLSS framework, the predictor vectors are modeled as linear combinations of covariates:

$$\vec{\eta}_k = \mathbf{X}_k \vec{\beta}_k, \tag{4.3}$$

where $\mathbf{X}_k \in \mathbb{R}^{N \times \tilde{P}_k}$ is the design matrix containing the covariate information for parameter k, with $\tilde{P}_k$ denoting the number of features used for modeling the distribution's k-th parameter, and $\vec{\beta}_k$ is the corresponding coefficient vector (Stasinopoulos et al., 2024, p. 23).[2] For an individual observation i, the predictor for the k-th parameter can be written as $\eta_{ik} = x_{i1k}\beta_{1k} + x_{i2k}\beta_{2k} + \dots + x_{i\tilde{P}_k k}\beta_{\tilde{P}_k k}$, where x_{ijk} represents the j-th covariate value for observation i used in modeling parameter k.

The basic GAMLSS framework presented above can be extended to incorporate more flexible and complex relationships by replacing linear predictors with structured additive predictors. This extension allows capturing non-linear relations between features and responses through non-linear function mappings, such as P-splines (Stasinopoulos et al., 2024, p. 22). The subsequent section introduces structured additive predictors in detail, followed by a discussion of Bayesian P-splines.

4.2 Structured Additive Predictors

The linear predictors in standard GAMLSS, as presented in the previous section, may be too restrictive to adequately capture complex relationships between covariates and the parameters of the response distribution. Structured additive predictors (Hastie & Tibshirani, 1986) extend these linear predictors, offering greater flexibility while maintaining interpretability (Stasinopoulos et al., 2024, pp. 21–22).

In the structured additive framework, each predictor $\vec{\eta}_k$ for parameter k is specified as an additive composition of D_k different effect types:

$$\eta_{ik} = \beta_{0k} + f_{k1}(\vec{\mathfrak{x}}_i) + \dots + f_{kJ_k}(\vec{\mathfrak{x}}_i), \tag{4.4}$$

for $i = 1,\dots,N$ and $k = 1,\dots,K$, where β_{0k} represents the overall level of the predictor (intercept term), and $f_{kj}(\vec{\mathfrak{x}}_i)$ for $j = 1,\dots,J_k$ denote different effect types applied to (potentially different subsets of) the

[2] For simplicity of notation, the $\mathbf{K}$ design matrices, which might include subsets of covariates, are assumed here to all consist of $\tilde{P}$ covariates.

covariate vector $\vec{\mathfrak{x}}_i$ (Stasinopoulos et al., 2024, p. 21).[3,4]

In vector notation, for each parameter k:

$$\vec{\eta}_k = \beta_{0k}\vec{1} + \vec{f}_{k1} + \ldots + \vec{f}_{kJ_k}, \tag{4.5}$$

where $\vec{1}$ is an N-dimensional vector of ones, and $\vec{f}_{kj} = (f_{kj}(\vec{\mathfrak{x}}_1), \ldots, f_{kj}(\vec{\mathfrak{x}}_N))^T$ is the vector of function evaluations for all observations.

This thesis defines functions $f_{kj}(\cdot)$ as linear combinations of B-spline basis functions and considers the univariate covariate case $\vec{x}$ with a single effect term, i. e. $J_k = 1$ for all k. For an observation i, the structured additive predictor of each parameter is a linear combination of D_k basis functions:

$$f_k(x_i) = \sum_{d=1}^{D_k} \gamma_{dk} B_{dk}(x_i), \tag{4.6}$$

where $\gamma_{dk} \in \mathbb{R}$ is a regression coefficient specific to the k-th parameter and $\text{Image}(f_k(x_i)) \in \mathbb{R}^{D_k}$.

For the vector of all observations $\vec{x}$ it applies that

$$f_k(\vec{x}) = \sum_{d=1}^{D_k} \gamma_{dk} B_{dk}(\vec{x}), \tag{4.7}$$

denotes that the B-spline basis functions are evaluated for each measurement in $\vec{x}$ and $\text{Image}(f_k(\vec{x})) \in \mathbb{R}^{N \times D_k}$.

In matrix notation, this can be written as:

$$f_k(\vec{x}) = \mathbf{Z}_k \vec{\gamma}_k, \tag{4.8}$$

where $\mathbf{Z}_k \in \mathbb{R}^{N \times D_k}$ is the design matrix contains the N B-spline basis transformation evaluations and $\vec{\gamma}_k = (\gamma_{1k}, \ldots, \gamma_{D_k k})^T \in \mathbb{R}^{D_k}$ is a vector of SAP regression coefficients for the k-th parameter (Stasinopoulos et al., 2024, p. 22).

The complete structured additive predictor can then be expressed as:

$$\vec{\eta}_k = \beta_{0k}\vec{1} + \mathbf{Z}_k \vec{\gamma}_k. \tag{4.9}$$

[3] In structured additive predictors, the generic notation f is used to denote a broad range of effect types, such as linear and non-linear effects of continuous covariates, tensor product interactions, spatial effects via Markov random fields, random effects, varying coefficient terms, regularized linear effects, or functional effects (Stasinopoulos et al., 2024, pp. 53–73).

[4] This thesis excludes interaction terms.

To make the model identifiable, appropriate centering constraints must be applied to the different functions (Stasinopoulos et al., 2024, p. 22). Furthermore, to account for potential overfitting and ensure desired smoothness properties, each coefficient vector $\overrightarrow{\gamma}_{kj}$ is associated with a regularization term, which in a Bayesian framework corresponds to defining a prior over the regression coefficients (Stasinopoulos et al., 2024, p. 23). These aspects will be addressed in detail in the subsequent subsections on Bayesian P-splines.

4.3 Bayesian Penalized B-Splines (P-Splines)

P-splines constitute an approach for modeling smooth non-linear relations between covariates and the response distribution parameters utilizing B-splines (Lang & Brezger, 2004). The Bayesian perspective dictates a smoothness penalty through a prior distribution on the B-spline basis coefficients, allowing the necessary degree of smoothness to be inferred by the data itself. This section describes the mathematical foundation of B-splines, details the prior specification for coefficient regularization, addresses identifiability constraints, and demonstrates the incorporation of P-splines within structured additive distributional regression models.

4.3.1 B-Spline Basis Functions

P-splines, introduced by Eilers and Marx (1996), combine B-spline basis functions with a penalty for coefficient differences to control smoothness. For a continuous one-dimensional covariate x_i, the effect function $f(x_i)$ is represented as a linear combination of D B-spline basis functions:

$$f(x_i) = \sum_{d=1}^{D} \gamma_d B_d(x_i), \tag{4.10}$$

where $B_d(x_i)$ denotes the d-th B-spline basis function evaluated at x_i, and γ_d is the corresponding coefficient.

B-splines of degree $\mathfrak{q}$ are defined recursively over a sequence of knots $t_1, t_2, \ldots, t_D$ that span the covariate range. Each B-spline basis function exhibits local support, being positive only in intervals formed by $\mathfrak{q}+2$ adjacent knots. The number of B-spline basis functions D relates to the number of interior knots $\mathfrak{m}$ through $D = \mathfrak{m} + \mathfrak{q} + 1$. This thesis employs equidistant placement of interior knots while other options are quantile-based or visually based knot placement strategies. For practical implementation of the recursive definition, $2\mathfrak{q}$ outer knots must be added to the interior knot sequence, i. e. $t_{t-l}, t_{t-l+1}\, t_1, \ldots, t_D, t_{D+l-1}, t_{D+l}$, by continuing the equidistant spacing outside the covariate domain (Fahrmeir et al., 2013, pp. 426–430).

The recursive algorithm for evaluating B-spline basis functions, a. k. a. the DE-BOOR algorithm, is defined

as:

$$B_d^0(x_i) = \begin{cases} 1 & \text{if } t_d \leq x_i < t_{d+1}, \\ 0 & \text{otherwise,} \end{cases} \tag{4.11}$$

$$B_d^{\mathfrak{p}}(x_i) = \frac{x_i - t_d}{t_{d+\mathfrak{p}} - t_d} B_d^{\mathfrak{p}-1}(x_i) + \frac{t_{d+\mathfrak{p}+1} - x_i}{t_{d+\mathfrak{p}+1} - t_{d+1}} B_{d+1}^{\mathfrak{p}-1}(x_i), \tag{4.12}$$

for $\mathfrak{p} = 1, 2, \ldots, \mathfrak{q}$, where $B_d^{\mathfrak{p}}(x_i)$ denotes a B-spline of degree $\mathfrak{p}$ (de Boor, 2001, pp.109–127; Fahrmeir et al., 2013, pp. 428–429). For notational convenience, it holds in the following that $B_d(x) = B_d^{\mathfrak{p}}(x)$.

4.3.2 Bayesian Regularization through Prior Specification

In a frequentist setting, smoothness is ensured by penalizing differences between adjacent coefficients (Eilers & Marx, 1996). Invented by Lang and Brezger, 2004, the Bayesian equivalent involves placing an informative Degenerate Multivariate Normal (DMN) global smoothness prior over the coefficient vector $\overrightarrow{\gamma}$ (Lang & Brezger, 2004), i. e.

$$p(\overrightarrow{\gamma} \mid \lambda^2) = \frac{1}{(2\pi\lambda^2)^{\frac{D-k}{2}} |\mathbf{K}|_+^{-\frac{1}{2}}} \exp\left[-\frac{1}{2\lambda^2} \overrightarrow{\gamma}^T \mathbf{K} \overrightarrow{\gamma} \right], \tag{4.13}$$

parameterized by a location vector, which is assumed to be a fixed zero-vector and thus omitted in (4.13), and a rank-deficient precision matrix $\frac{1}{\lambda^2}\mathbf{K}$ (Fahrmeir et al., 2013, p. 41; Fahrmeir et al., 2013, pp. 650–651; Holbrook, 2018).[5]

The matrix $\mathbf{K}$ defines the smoothness properties and is constructed as $\mathbf{K} = \mathbf{D}^T\mathbf{D}$, where $\mathbf{D}$ is a difference matrix of order k. This thesis uses a difference matrix of order $k = 2$, i. e. the difference matrix $\mathbf{D} \in \mathbb{R}^{(D-2)\times D}$ is defined as:

$$\mathbf{D} = \begin{pmatrix} 1 & -2 & 1 & 0 & \cdots & 0 \\ 0 & 1 & -2 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 1 & -2 & 1 \end{pmatrix}. \tag{4.14}$$

Thus, given an order k, the matrix $\mathbf{K}$ with $\mathbb{R}^{D\times D}$ has $\text{rank}(\mathbf{K}) = D - k$ (Fahrmeir et al., 2013, p. 437; Fahrmeir et al., 2013, p. 443).

The variance parameter $\lambda^2 \in \mathbb{R}_+$ controls the balance between smoothness and flexibility, acting as an inverse smoothing parameter. As $\lambda^2 \to 0$, the function approaches a polynomial of degree $k-1$, while larger values of λ^2 allow for more flexibility (Lang & Brezger, 2004).

[5] A derivation of the DMN is provided in Appendix A.1.

4.3.3 Hyperprior over the Variance Parameter

From a Bayesian perspective, λ^2 is treated as a random variable (Lang & Brezger, 2004). Various weakly informative hyperpriors[6], such as an Inverse Gamma (IG), a Half-Cauchy, or a Half-Normal prior (Klein & Kneib, 2016), can be specified over λ^2, which allows for data-driven smoothing.

Following Lang and Brezger (2004), this thesis chooses the IG as a hyperprior:

$$\lambda^2 \sim \text{IG}(a, b) \tag{4.15}$$

where a, the concentration and b, the scale, are hyperparameters. Common choices include $a = 1$ and $b \in \{0.005, 0.0005, 0.00005\}$ for a weakly informative prior (Lang & Brezger, 2004).

4.3.4 Identifiability Constraints

The identifiability problem in additive regression models, where components can be shifted by constants without changing the overall model fit, i. e. $f_{p,1}(x_i, \gamma_{p,1}) + f_{p,2}(x_i, \gamma_{p,2}) = f_{p,1}(x_i, \gamma_{p,1}) + c + f_{p,2}(x_i, \gamma_{p,2}) - c$, is resolved by imposing a sum-to-zero constraint (Hastie & Tibshirani, 1986).[7] This constraint is enforced by transforming the design matrix $\mathbf{Z}$ by projecting it onto a space orthogonal to constant shifts, followed by corresponding adjustments to the penalty matrix $\mathbf{K}$, ensuring that the additive components sum to zero across observations (Wood, 2017, p. 45; Wood, 2017, pp. 163–164). Applying the constraint has the consequence that one dimension in $\vec{\gamma}$ is lost. For a detailed presentation of applying the sum-to-zero constrained in the context of P-splines, please refer to Callegher et al. (2025).

For ease of notation in this thesis, it is assumed that references to the design matrix $\mathbf{Z}$ and the penalty matrix $\mathbf{K}$ within the framework of P-splines always relate to their constrained versions.

4.3.5 Structured Additive Distributional Regression with Bayesian P-Splines

By integrating P-splines within the structured additive predictor framework, the SADR model (Callegher et al., 2025; Kleinemeier & Klein, 2023) is formulated as:

$$\vec{y} \overset{\text{ind.}}{\sim} p(\vec{y} \mid \vec{\theta}_1, \ldots, \vec{\theta}_K), \tag{4.16}$$

$$\vec{\theta}_k = g_k^{-1}(\vec{\eta}_k) \quad \text{for } k = 1, \ldots, K, \tag{4.17}$$

$$\vec{\eta}_k = \beta_{0k}\vec{1} + \mathbf{Z}_k\vec{\gamma}_k, \tag{4.18}$$

with priors:

[6] In Bayesian hierarchical modeling a prior in that hierarchy level is termed a hyperprior (Gelman et al., 2014, pp. 107–133).

[7] While the sum-to-zero constraint is commonly applied in generalized additive models, alternative approaches like point constraints also exist, and the choice between these constraints impacts the standard errors of the estimated splines (Stringer, 2023).

$$\beta_{0k} \sim N(\mu_{0k}, \sigma_{0k}), \tag{4.19}$$

$$\overrightarrow{\gamma}_k \mid \lambda_k^2, \mathbf{K}_k \sim \text{DMN}(\overrightarrow{0}, \frac{\mathbf{K}_k}{\lambda_k^2}), \tag{4.20}$$

$$\lambda_k^2 \sim \text{IG}(a_k, b_k), \tag{4.21}$$

for each parameter k with μ_{0k} and σ_{0k} being the location and scale, respectively (Callegher et al., 2025; Lang & Brezger, 2004).

4.4 Application to the Simulation Study and the Case Study's Youngman Two-Stage Approach

The Bayesian SADR framework presented in the previous sections can be applied to various parametric distributions. This section addresses the application to two distributions central to Youngman's two-stage approach for extreme value modeling (Youngman, 2019): the ALD for quantile regression in the first stage, and the GPD for modeling exceedances over the threshold in the second stage. The Bayesian SADR framework for the GPD involved in the simulation study is specified analogously for the case study. Figure 4.1 represents the probabilistic model specified for the two-stage approach in plate notation.

4.4.1 SADR Modeling Implications for PoT

Chapter 2 establishes that for a sequence of N independent and identical random variables $(Y_i)_1^N$ with common distribution function F, the conditional distribution of excesses

$$Y_i - u | Y_i > u,$$

converges against a GPD for a sufficiently high threshold u. The SADR framework modifies this in two ways: First, SADR assumes N independent but not necessarily identical random variables $\{Y_i\}_1^N$ that follow a common distribution function F. Second, when applying SADR for Youngman's two-stage PoT approach, the threshold u is no longer constant but is obtained by inferring the ALD's location parameter given a measured covariate x_i for observation i. Consequently, the conditional distribution of excesses

$$Y_i - u_\tau(x_i) | Y_i > u_\tau(x_i),$$

converges against a GPD for a sufficiently high threshold $u_\tau(x_i)$.

4.4.2 Asymmetric Laplace Distribution

The ALD is parameterized as $\text{AL}(y_i|u_\tau, \psi)$, where $u_\tau \in \mathbb{R}$ is the location parameter, $\psi \in \mathbb{R}_+$ is the scale parameter, and $\tau \in (0,1)$ is the fixed quantile-level specified by the user (Yu & Moyeed, 2001).

Within the SADR framework, the location parameter u_τ and the scale parameter ψ are linked to the structured additive predictors η_u and η_ψ, respectively, i. e.

$$u_\tau(\vec{x}) = g_u^{-1}(\vec{\eta_u}), \tag{4.22}$$
$$\psi(\vec{x}) = g_\psi^{-1}(\vec{\eta_\psi}), \tag{4.23}$$

where g_u^{-1} is the identity response function for the location parameter, g_ψ^{-1} is a response function that ensures positivity, and both response functions are applied element-wise. The structured additive predictors are specified as:

$$\vec{\eta}_u = \beta_{0u} + \mathbf{Z}^{\text{AL}} \vec{\gamma}_u, \tag{4.24}$$
$$\vec{\eta}_\psi = \beta_{0\psi} + \mathbf{Z}^{\text{AL}} \vec{\gamma}_\psi, \tag{4.25}$$

using the identical design matrix $\mathbf{Z}^{\text{AL}}$ for both parameter predictors, as this thesis uses the same univariate covariate for each, with the superscript AL indicating that the design matrix is associated with the AL distributed responses. For the ALD, the design matrix $\mathbf{Z}^{\text{AL}}$ contains the N P-spline constrained B-spline basis transformations evaluated for the covariate $\vec{x}$.

Consequently, it is assumed that $\vec{Y} \overset{\text{ind.}}{\sim} \text{AL}(u_\tau(\vec{x}), \psi(\vec{x}))$.[8]

The ALD is used for quantile regression to estimate a covariate-dependent threshold $u_\tau(\vec{x})$ at a specified quantile-level τ.

Following Section 4.3.5, the Bayesian specification is completed with appropriate priors for the SAP parameters:

$$\begin{aligned} \beta_{0u} &\sim \text{N}(\mu_u, \sigma_u), \quad \lambda_u^2 \sim \text{IG}(a_u, b_u), \quad \vec{\gamma}_u \sim \text{DMN}(\vec{0}, \tfrac{\mathbf{K}^{\text{AL}}}{\lambda_u^2}), \\ \beta_{0\psi} &\sim \text{N}(\mu_\psi, \sigma_\psi), \quad \lambda_\psi^2 \sim \text{IG}(a_\psi, b_\psi), \quad \vec{\gamma}_\psi \sim \text{DMN}(\vec{0}, \tfrac{\mathbf{K}^{\text{AL}}}{\lambda_\psi^2}), \end{aligned} \tag{4.26}$$

with the identical penalty matrix $\mathbf{K}^{\text{AL}}$ for both parameter predictors, as they both rely on the design matrix $\mathbf{Z}^{\text{AL}}$ and the same B-spline configuration, as detailed in Chapter 6.

The chosen response function and hyperparameter values are detailed in Chapter 6.

[8] The notation $\vec{Y} \sim \text{AL}(u_\tau(\vec{x}), \psi(\vec{x}))$ does not denote a multivariate Asymmetric Laplace Distribution. Rather, it should be understood that each element Y_i in $\vec{Y}$ is independently AL distributed with the corresponding elements $u_\tau(x_i)$ and $\psi(x_i)$ from the vectors $u_\tau(\vec{x})$ and $\psi(\vec{x})$ as its parameters, i. e. $Y_i \sim \text{AL}(u_\tau(x_i), \psi(x_i))$ for all $i = 1, 2, \ldots, N$.

4.4.3 Generalized Pareto Distribution

The GPD is parameterized as $\text{GP}(y_i|\sigma,\xi)$, where $\sigma \in \mathbb{R}_+$ is the scale parameter and $\xi \in \mathbb{R}$ is the shape parameter. Let $\overrightarrow{y}^{\text{GP}} = (y_\mathfrak{l} - u_\tau(x_\mathfrak{l}))_{i\in I}$ with $I = \{i : y_i > u_\tau(x_i), i = 1,2,\ldots,N\}$ represent the vector of all y_i elements exceeding $u_\tau(x_i)$, centered by subtracting $u_\tau(x_\mathfrak{l})$ from each exceedance, with $|\overrightarrow{y}^{\text{GP}}| = M$ as the number of exceedances. Within the SADR framework, the scale parameter σ and the shape parameter ξ are linked to the structured additive predictors η_σ and η_ξ, respectively, i. e.

$$\sigma(\overrightarrow{x}^{\text{GP}}) = g_\sigma^{-1}(\overrightarrow{\eta}_\sigma), \tag{4.27}$$

$$\xi(\overrightarrow{x}^{\text{GP}}) = g_\xi^{-1}(\overrightarrow{\eta}_\xi), \tag{4.28}$$

where $\overrightarrow{x}^{\text{GP}} = (x_\mathfrak{l})_{\mathfrak{l}\in I}$ with the same index set I denotes the vector of all x_i for the corresponding response $y_i > u_\tau(x_i)$ that exceeds the threshold. The function g_ξ^{-1} is the identity response function for the shape parameter, g_σ^{-1} is a response function that ensures positivity, and both response functions are applied element-wise. The structured additive predictors are specified as:

$$\overrightarrow{\eta}_\sigma = \beta_{0\sigma} + \mathbf{Z}^{\text{GP}}\overrightarrow{\gamma}_\sigma, \tag{4.29}$$

$$\overrightarrow{\eta}_\xi = \beta_{0\xi} + \mathbf{Z}^{\text{GP}}\overrightarrow{\gamma}_\xi, \tag{4.30}$$

with the identical design matrix $\mathbf{Z}^{\text{GP}}$ for both parameter predictors since this thesis uses the same centered exceedances over the threshold for both of them. The superscript GP indicates that the design matrix is associated with the GP distributed responses. For the GPD, the design matrix $\mathbf{Z}^{\text{GP}}$ contains the N P-spline constrained B-spline basis transformations evaluated for each element in the vector $\overrightarrow{x}^{\text{GP}}$, i. e. the GPD scale $\sigma(\overrightarrow{x}^{\text{GP}})$ and shape $\xi(\overrightarrow{x}^{\text{GP}})$ parameters are modeled as functions of the exceedances' covariates.

Consequently, it is assumed that $\overrightarrow{Y}^{\text{GP}} = (Y_\mathfrak{l} - u_\tau(x_\mathfrak{l}))_{\mathfrak{l}\in I}$ is the vector of all threshold exceeding random variables centered around the threshold. These random variables are assumed to be GP-distributed, i. e. $\overrightarrow{Y}^{\text{GP}} \overset{\text{ind.}}{\sim} \text{GP}(\sigma(\overrightarrow{x}^{\text{GP}}),\xi(\overrightarrow{x}^{\text{GP}}))$.[9]

Following Subsection 4.3.5, the Bayesian specification is completed with appropriate priors for the SAP parameters:

$$\begin{aligned} \beta_{0\sigma} &\sim \text{N}(\mu_\sigma,\sigma_\sigma), \quad \lambda_\sigma^2 \sim \text{IG}(a_\sigma,b_\sigma), \quad \overrightarrow{\gamma}_\sigma \sim \text{DMN}(\overrightarrow{0},\tfrac{\mathbf{K}^{\text{GP}}}{\lambda_\sigma^2}), \\ \beta_{0\xi} &\sim \text{N}(\mu_\xi,\sigma_\xi), \quad \lambda_\xi^2 \sim \text{IG}(a_\xi,b_\xi), \quad \overrightarrow{\gamma}_\xi \sim \text{DMN}(\overrightarrow{0},\tfrac{\mathbf{K}^{\text{GP}}}{\lambda_\xi^2}), \end{aligned} \tag{4.31}$$

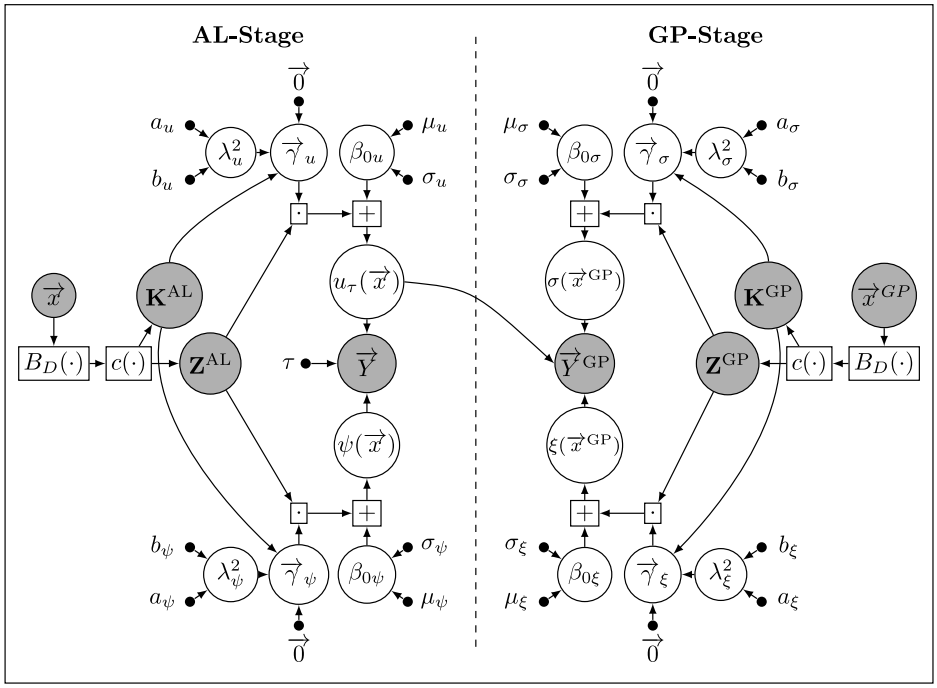
with the identical penalty matrix $\mathbf{K}^{\text{AL}}$ for both parameter predictors, as they both rely on the design matrix $\mathbf{Z}^{\text{AL}}$ and the same B-spline configuration, as detailed in Chapter 6.

The GPD model specification for the simulation study is analogous to the case study, but with the

[9] The notation $\overrightarrow{Y}^{\text{GP}} \sim \text{GP}(\sigma(\overrightarrow{x}^{\text{GP}}),\xi(\overrightarrow{x}^{\text{GP}}))$ does not denote a multivariate Generalized Pareto Distribution. Rather, it should be understood that random variable $Y_\mathfrak{l}$ in $\overrightarrow{Y}^{\text{GP}}$ is independently GP distributed with the corresponding elements $\sigma(x_\mathfrak{l})$ and $\xi(x_\mathfrak{l})$ from the vectors $\sigma(\overrightarrow{x}^{\text{GP}})$ and $\xi(\overrightarrow{x}^{\text{GP}})$ as its parameters, i. e. $Y_\mathfrak{l} \sim \text{GP}(\sigma(x_\mathfrak{l}),\xi(x_\mathfrak{l}))$ for all $\mathfrak{l} \in I$.

difference that no threshold has to be determined since the simulated data is sampled from a GPD with a known threshold (see Chapter 6).

The chosen response function and hyperparameter values are detailed in Chapter 6.



Symbol	Description
•	Hyperparameter
○	Unobserved random variable
◍	Observed variable
$B_D(\cdot)$	B-spline basis transformation
$\boxplus$	Element-wise addition
$\boxdot$	Matrix-Vector product
$c(\cdot)$	P-spline constraint transformation

Figure 4.1: Plate Notation for the Probabilistic Two-Stage PoT Approach.
This plate notation is also applicable to the simulation study by cutting the link between the GP-Stage and the AL-Stage and replacing the $u_\tau(\overrightarrow{x})$ node with a zero-vector hyperparameter node (refer to Chapter 6 for details about the simulation study's configuration).

Chapter 5

Variational Inference

VI frames the Bayesian problem of inferring the posterior as an optimization problem (Attias, 1999; Ghahramani & Beal, 2000; Hinton & van Camp, 1993; Jaakkola & Jordan, 2000; Jordan et al., 1999; MacKay, 1995; Opper & Winther, 1996; Wainwright & Jordan, 2007). VI is typically employed when the evidence integral is intractable to compute, using optimization techniques to approximate posterior distributions rather than drawing samples as in MCMC methods. Although MCMC methods provide asymptotically exact posterior samples, they scale poorly to massive datasets. While VI offers significantly faster computation than MCMC, this is at the cost of potentially underestimating posterior variance and providing no guarantees of recovering the exact posterior (Blei et al., 2017). Its roots are in the calculus of variations which objects to finding stationary points of functionals, specifically extrema, with the search domain $\mathcal{Q}$ being a set of admissible functions (Courant & Hilbert, 1989, p. 167).

Let $q(\vec{\theta}) \in \mathcal{Q}$ be a probability distribution from a set of admissible probability distributions $\mathcal{Q}$ (named the variational family) over the vector of latent variables $\vec{\theta} \in \Theta \in \mathbb{R}^m$ with Θ being an m-dimensional latent space, let $p(\vec{\theta}\,|\mathcal{D})$[1] denote the posterior over the latent variables $\vec{\theta}$ given data $\mathcal{D}$, and let the KL divergence (Kullback & Leibler, 1951)

$$\mathrm{KL}(q(\vec{\theta})||p(\vec{\theta}\,|\mathcal{D})) = \int q(\vec{\theta}) \ln\left(\frac{q(\vec{\theta})}{p(\vec{\theta}\,|\mathcal{D})}\right) d\vec{\theta}\,, \tag{5.1}$$

quantify the divergence from the variational distribution $q(\vec{\theta})$ to the posterior $p(\vec{\theta}\,|\mathcal{D})$ with the KL divergence being non-negative and 0 if and only if the two distributions are equal, i. e.

$$\mathrm{KL}(q(\vec{\theta})||p(\vec{\theta}\,|\mathcal{D})) = 0 \iff q(\vec{\theta}) \equiv p(\vec{\theta}\,|\mathcal{D})$$

[1] For notational convenience this thesis omits denoting the hyperparameter vector $\vec{\kappa}$ in the following. With hyperparameters the notation for the posterior, evidence, likelihood, and joint distribution would be $p(\vec{\theta}\,|\mathcal{D}, \vec{\kappa})$, $p(\mathcal{D}|\vec{\kappa})$, $p(D|\vec{\theta}\,, \vec{\kappa})$, and $p(D, \vec{\theta}\,|\vec{\kappa})$, respectively.

. Then, VI has the objective to minimize the KL divergence w. r. t. the variational distribution, i. e.

$$p(\vec{\theta}\,|\mathscr{D}) = \arg\min_{q(\vec{\theta})\in\mathscr{Q}} \mathrm{KL}(q(\vec{\theta})||p(\vec{\theta}\,|\mathscr{D})), \tag{5.2}$$

with the KL divergence being a functional and $q(\vec{\theta})$ being a probability distribution from the variational family $\mathscr{Q}$ over whose domain the minimum is searched for using calculus of variations.

Typically, the KL divergence is not minimized, but the negative ELBO (derived by Jordan et al. (1999)) that equals the negative logarithmic evidence plus the KL divergence between the approximate posterior $q(\vec{\theta})$ and the posterior $p(\vec{\theta}\,|\mathscr{D})$, i. e.

$$\begin{aligned}
\mathrm{KL}(q(\vec{\theta})||p(\vec{\theta}\,|\mathscr{D})) &= \int q(\vec{\theta}) \ln\left(\frac{q(\vec{\theta})}{p(\vec{\theta}\,|\mathscr{D})}\right) d\vec{\theta} \\
&= \mathbb{E}_{q(\vec{\theta})}[\ln q(\vec{\theta})] - \mathbb{E}_{q(\vec{\theta})}[\ln p(\vec{\theta}\,|\mathscr{D})] \\
&= \mathbb{E}_{q(\vec{\theta})}[\ln q(\vec{\theta})] - \mathbb{E}_{q(\vec{\theta})}\left[\ln\left(p(\vec{\theta}\,|\mathscr{D})\frac{p(\mathscr{D})}{p(\mathscr{D})}\right)\right] \\
&= \mathbb{E}_{q(\vec{\theta})}[\ln q(\vec{\theta})] - \mathbb{E}_{q(\vec{\theta})}\left[\ln\left(p(\vec{\theta},\mathscr{D})\frac{1}{p(\mathscr{D})}\right)\right] \\
&= \mathbb{E}_{q(\vec{\theta})}[\ln q(\vec{\theta})] - \mathbb{E}_{q(\vec{\theta})}[\ln p(\vec{\theta},\mathscr{D})] + \mathbb{E}_{q(\vec{\theta})}[\ln p(\mathscr{D})] \\
&= \mathbb{E}_{q(\vec{\theta})}[\ln q(\vec{\theta})] - \mathbb{E}_{q(\vec{\theta})}[\ln p(\vec{\theta},\mathscr{D})] + \ln p(\mathscr{D}) \\
\Leftrightarrow -\ln p(\mathscr{D}) &= \mathbb{E}_{q(\vec{\theta})}[\ln q(\vec{\theta})] - \mathbb{E}_{\vec{\theta}\sim q(\vec{\theta})}[\ln p(\vec{\theta},\mathscr{D})] - \mathrm{KL}(q\,||\,p) \quad \Big|\cdot(-1) \\
\Leftrightarrow \quad \ln p(\mathscr{D}) &= -\mathbb{E}_{q(\vec{\theta})}[\ln q(\vec{\theta})] + \mathbb{E}_{q(\vec{\theta})}[\ln p(\vec{\theta},\mathscr{D})] + \mathrm{KL}(q\,||\,p) \\
&= \mathbb{E}_{q(\vec{\theta})}\left[\frac{\ln p(\vec{\theta},\mathscr{D})}{\ln q(\vec{\theta})}\right] + \mathrm{KL}(q\,||\,p) \\
&= \mathrm{ELBO}(q(\vec{\theta})) + \mathrm{KL}(q(\vec{\theta})||p(\vec{\theta}\,|\mathscr{D})) \quad \Big|-\ln p(\mathscr{D}); -\mathrm{ELBO}(q(\vec{\theta})) \\
\Leftrightarrow -\mathrm{ELBO}(q(\vec{\theta})) &= \mathrm{KL}(q(\vec{\theta})||p(\vec{\theta}\,|\mathscr{D})) - \ln p(\mathscr{D})
\end{aligned} \tag{5.3}$$

(Blei et al., 2017).

Given (5.3) and that the evidence $p(\mathscr{D})$ does not depend on the latent variables $\vec{\theta}$, it holds that

$$\arg\min_{q(\vec{\theta})\in\mathscr{Q}} \mathrm{KL}(q(\vec{\theta})||p(\vec{\theta}\,|\mathscr{D})) \equiv \arg\min_{q(\vec{\theta})\in\mathscr{Q}} -\mathrm{ELBO}(q(\vec{\theta})), \tag{5.4}$$

(Blei et al., 2017).

Although calculus of variations is not inherently an approximation method, VI is an approximation method, as it needs to restrict the variational family $\mathscr{Q}$ (Bishop, 2006, p. 463) with the approximation accuracy, thus, depending on the chosen variational family (Blei et al., 2017).

Nakajima et al. (2019, p. 40) assert that the evaluation of ELBO often presents challenges for most probability distributions due to the expectation over $q(\vec{\theta})$ and therefore requires constraining the variational family $\mathcal{Q}$ of all probability distributions to a variational family with tractable probability distributions. A classic variational family of choice is the mean-field variational family which assumes that the latent variables are mutually independent and can thus be factorized, i. e.

$$q(\vec{\theta}) = \Pi_{j=1}^{m} q_j(\theta_j), \tag{5.5}$$

with θ_j being an element of $\vec{\theta}$ and q_j not necessarily being the same distributions for all j (Blei et al., 2017). The mean field variational family is detailed in Section 5.1 and shall also motivate the necessity of stochastic VI.

While the mean-field variational family allows for inferring the posterior in some modeling scenarios, it does not for all. Another commonly used restriction to the variational family $\mathcal{Q}$ is to set the function space to a parametric probability distribution family $\mathcal{P}_{\vec{\phi}}$ with $\vec{\phi} \in \Phi \in \mathbb{R}^c$ which reduces the optimization over the functional space to an unconstrained minimization in the parameter space Φ, i. e.

$$p(\vec{\theta}\,|\mathcal{D}, \vec{\phi}) = \arg\min_{\vec{\phi}\in\Phi} -\mathrm{ELBO}(\vec{\theta}, \vec{\phi}), \tag{5.6}$$

with the ELBO functional becoming a function of the latent variables $\vec{\theta}$ and the variational parameters $\vec{\phi}$ (Nakajima et al., 2019, p. 47). This variational family is used for stochastic VI which will be detailed in Section 5.2.[2]

5.1 Mean-Field Variational Inference

VI using a mean-field variational family which assumes that the latent variables are mutually independent (see (5.5)) is commonly called Mean-Field Variational Inference (MFVI) (Blei et al., 2017).

MFVI does not guarantee an optimal solution to the objective (5.4) in general. However, optimality can be achieved under specific model assumptions. For example, when the model $p(\mathcal{D}, \vec{\theta})$ contains conditionally conjugate priors that can be factorized as $p(\vec{\theta}) = \prod_j^m p_j(\theta_j)$, stationary conditions can be derived using the calculus of variations. Under this conditionally conjugate model prior assumption, a stationary condition exists for each factor q_j. This stationary condition for q_j can be expressed explicitly in parametric form as an expectation over the r latent variables $r \neq j$, i. e.

$$0 \overset{!}{=} \frac{\partial -\mathrm{ELBO}}{\partial q_j} = \mathbb{E}_{\theta_{r\neq j}\sim q(\theta_{r\neq j})}\left[\ln p(\mathcal{D}, \vec{\theta})\right] + \text{constant}, \tag{5.7}$$

with the expectation being known for conditionally conjugate models (Nakajima et al., 2019, pp. 41–46).

[2]The here presented restricted variational families are not exhaustive, e. g. there is also the structured variational family which expands the mean-field variational family by adding dependencies (Barber & Wiegerinck, 1998; Saul & Jordan, 1995).

Assuming the model and that the user has derived the explicit form of the expectations in (5.7), the optimal j-th latent variable depends on the other latent variables whose optimum also needs to be determined. The Coordinate Ascend Variational Inference (CAVI) algorithm iteratively optimizes these interdependent parameters (see algorithm 1). CAVI converges to a local optimum when ELBO(q) is non-convex and to a global optimum when it is convex (Blei et al., 2017).

Algorithm 1 CAVI (Blei et al., 2017)

Input: A model $p(\mathscr{D}, \vec{\theta})$, a data set $\mathscr{D}$
Output: An optimal variational density $q^*(z) = \prod_{j=1}^{m} q_j^*(\vec{\theta}_j)$
Initialize: Variational factors $q_j(\vec{\theta}_j)$
while the ELBO has not converged **do**
    for $j \in \{1, \ldots, m\}$ **do**
        Set $q_j(\vec{\theta}_j) \propto \exp\left\{\mathbb{E}_{\vec{\theta}_{i \neq j}}\left[\ln p(X, \vec{\theta})\right]\right\}$
    end for
    Compute ELBO($\vec{\theta}$)
end while
return $q^*(\vec{\theta})$

MFVI constructs the best approximation of the unknown posterior distribution $p(\vec{\theta}|\mathscr{D})$ within the variational family of the mean field, but it has some drawbacks. Deriving the functional form of optimal variational factors requires a manual search rather than a computational search for the solution, which can be tedious for complex models. In addition, a solution for the optimal functional in equation (5.7) is not guaranteed for all statistical models. Some modeling choices can provide guarantees, but otherwise MFVI offers limited model flexibility to users (Paisley et al., 2012; Ranganath et al., 2014). A more flexible VI framework is provided by stochastic VI.

5.2 Stochastic Variational Inference

SVI addresses limitations of MFVI, particularly the restricted model choice and the difficulties in deriving optimal analytical solutions (Paisley et al., 2012; Ranganath et al., 2014). SVI has become more popular over Mean Field VI in recent years (Sjölund, 2023). It differs from MVFI by restricting the variational family $\mathscr{Q}$ to a parametric probability distribution family which induces the objective (5.6), i. e. the negative ELBO is seen as a function dependent on the latent variables $\vec{\theta}$ and the variational parameters $\vec{\phi}$ and the search space is the parameter space Φ.

A common approach to solving the objective in equation (5.6) for a given variational family $\mathscr{P}_{\vec{\phi}}$ involves gradient-based optimization with respect to the variational parameters (Homan & Gelman, 2014; Kingma & Welling, 2013; Kucukelbir et al., 2016; Liu & Wang, 2016; Ranganath et al., 2014; Roeder et al., 2017). Gradient-based optimization employs iterative parameter updates based on unbiased but noisy gradient estimates—a technique known as SGD when updating the parameters in the negative gradient direction

to minimize the objective function and Stochatic Gradient Ascent (SGA) when updating the parameters in the positive gradient direction to maximize the objective function. The method converges to a local optimum of the objective function when appropriate step sizes are selected (Robbins & Monro, 1951).

Various approaches for estimating the ELBO's gradients w. r. t. the variational parameters exist, e. g. Hoffman et al. (2013) estimate the ELBO's natural gradients by estimating them from a subset of the dataset (mini-batches), Ranganath et al. (2014) estimate the ELBO's gradients via the score gradient estimator (see Appendix B.1), and Kingma and Welling (2013) estimate the ELBO's gradients with the reparameterization gradient estimator.

Reparameterization Gradient Estimator: Let $\vec{\epsilon}$ be a auxiliary random variable distributed by an auxiliary distribution $p(\vec{\epsilon})$. Assume the existence of a continuously differentiable, injective function $g_{\vec{\phi}}$ parameterized by the variational parameters $\vec{\phi}$ with a non-vanishing Jacobian $J_{g_{\vec{\phi}}}$, such that

$$g_{\vec{\phi}} : \vec{\epsilon} \mapsto \vec{\theta},$$

and

$$g_{\vec{\phi}}^{-1} : \vec{\theta} \to \vec{\epsilon},$$

defined over the Image(g). Specifically, for a sample s, one has

$$g : \vec{\epsilon}^{\,s} \mapsto g_{\vec{\phi}}(\vec{\epsilon}) = \vec{\theta}^{\,s}$$

with $s = 1, ..., S$. It is the change of variables theorem for probability densities (Fahrmeir et al., 2013, p. 645) that allows the expression of samples of the latent variables $\vec{\theta}^{\,s}$ from the probability distribution $q(\vec{\theta} \mid \vec{\phi})$ as a reparameterization of random samples from $\vec{\epsilon}$ through $g_{\vec{\phi}}(\vec{\epsilon})$. For instance, if $q(\vec{\theta})$ is assumed to be a location-scale distribution with $\vec{\phi} = \{\text{location}, \text{scale}\}$, then for $\vec{\epsilon} \sim \mathcal{N}(\vec{0}_c, I_c)$, a sample s from $q(\vec{\theta})$ is given by $\vec{\theta}^{\,s} = \text{location} + \text{scale} \cdot \vec{\epsilon}^{\,s}$. Given a sample $\vec{\epsilon}^{\,s}$, $\vec{\theta}^{\,s}$ can then be considered deterministic. The randomness then stems from $\vec{\epsilon}$ and not from $\vec{\theta}$ anymore (Kingma & Welling, 2013; Kucukelbir et al., 2016).

Kingma and Welling, 2013 show that reparameterizing the latent variables allows to rewrite the ELBO's expected value integral as a Monte Carlo (MC) integration (Murphy, 2023, pp. 477–478):

$$\begin{aligned}
\text{ELBO}(\vec{\theta}, \vec{\phi}) &= \mathbb{E}_{\vec{\theta} \sim q(\vec{\theta} \mid \vec{\phi}))} \left[\ln \frac{p(\mathscr{D}, \vec{\theta})}{q(\vec{\theta} \mid \vec{\phi}))} \right] && (5.8) \\
&= \mathbb{E}_{\epsilon \sim p(\vec{\epsilon})} \left[\ln \frac{p(\mathscr{D}, g_{\vec{\phi}}(\vec{\epsilon}) \mid \vec{\theta})}{q(g_{\vec{\phi}}(\vec{\epsilon}))} \right] && (5.9) \\
&\overset{MC}{\approx} \frac{1}{S} \sum_{s=1}^{S} \ln \frac{p(\mathscr{D}, \vec{\theta}^{\,s})}{q(\vec{\theta}^{\,s} \mid \vec{\phi}))} \quad \text{with } \vec{\theta}^{\,s} = g_{\vec{\phi}}(\vec{\epsilon}^{\,s}), \vec{\epsilon} \sim p(\vec{\epsilon}). && (5.10)
\end{aligned}$$

Building on (5.10), the reparameterization trick allows to estimate the ELBO's gradients by taking the sample mean of the gradients of the log joint probability distribution and log variational probability distribution w. r. t. the reparameterized samples of the latent variables:

$$\begin{aligned}\nabla_{\phi}\,\mathrm{ELBO}(\vec{\theta}\,,\vec{\phi}) &\overset{MC}{\approx} \nabla_{\vec{\phi}}\,\frac{1}{S}\sum_{s=1}^{S}\ln\frac{p(\mathscr{D},\vec{\theta}^{\,s})}{q(\vec{\theta}^{\,s}\mid\vec{\phi})}\\ &= \frac{1}{S}\sum_{s=1}^{S}\nabla_{\vec{\phi}}\ln\frac{p(\mathscr{D},\vec{\theta}^{\,s})}{q(\vec{\theta}^{\,s}\mid\vec{\phi})}\,. \end{aligned} \tag{5.11}$$

Researchers in SVI have sought to reduce the variance of reparameterization gradient estimators for the $\mathrm{ELBO}(\vec{\theta}\,,\vec{\phi})$, since these estimators, while unbiased, are not necessarily efficient. For some models, the reparameterization gradient estimator empirically exhibits lower variance than other estimators like the score gradient estimator, but a general theoretical explanation remains difficult to attain, except for limited scenarios (Domke et al., 2023; Xu et al., 2018). One modification to reduce the variance of the reparameterization trick gradient estimator for some scenarios has been made by Roeder et al. (2017) with the Stick-the-Landing gradient estimator (see Appendix C.1).

Automatic Differentiation for SVI: Kucukelbir et al. (2016) demonstrate the use of automatic differentiation to compute the gradients of the ELBO w. r. t. the variational parameters $\vec{\phi}$ for SGD while implicitly employing the reparameterization gradient estimator. The ELBO gradient estimation process using automatic differentiation consists of drawing S reparameterized latent variable samples from the variational distribution given the current variational parameters $\vec{\phi}$, evaluating the ELBO for each sample, calculating the sample mean across these S draws, and then applying automatic differentiation to this result, i. e. automatic differentiation is applied to

$$\frac{1}{S}\sum_{s=1}^{S}\mathrm{ELBO}(\vec{\theta}\,,\vec{\phi}) = \frac{1}{S}\sum_{s=1}^{S}\left[\ln\frac{p(\mathscr{D},\vec{\theta}^{\,s})}{q(\vec{\theta}^{\,s}|\vec{\phi})}\right]. \tag{5.12}$$

The reparameterization trick enables gradient flow through the variational parameters, which would for example not be impossible with the score gradient estimator since sampling operations are non-differentiable (Kingma & Welling, 2013).

So far, for this approach to be possible, five requirements must be met: (1) the variational distribution $q(\cdot)$ can be evaluated, (2) the variational distribution $q(\cdot)$ is differentiable, (3) sampling from the variational distribution $q(\cdot)$ is reparameterizable, (4) the joint distribution $p(\mathscr{D},\vec{\theta}\,)$ can be evaluated, and (5) the joint distribution $p(\mathscr{D},\vec{\theta}\,)$ is differentiable (Kingma & Welling, 2013; Kucukelbir et al., 2016).

The following section introduces the selected variational distribution.

SVI with a Full Covariance Multivariate Normal Variational Distribution

This thesis employs a specifically parameterized Multivariate Normal distribution with a fully specified covariance matrix as the chosen variational distribution. To differentiate this distribution from both a Multivariate Normal distribution with a diagonal covariance matrix (where only variances but not covariances are specified) and from the (Multivariate) Normal distributions used as prior distributions in Chapter 4, the distribution is referred to as the FCMN distribution. The parameterization is taken up at the end of the subsection.

Since the latent variables $\vec{\theta}$ are not necessarily from $\mathbb{R}^m$ as does the support of the FCMN distribution, Kucukelbir et al. state that another requirement is the selection of appropriate transformations according to the latent space defined by the model. For this, a bijective and differentiable function T transforms a constrained latent variable $\vec{\zeta}$ from a constrained latent space $\Lambda \subseteq \mathbb{R}^c$ to the unconstrained latent space Θ, i. e.

$$\begin{aligned} T &: \Lambda \to \Theta \\ \vec{\zeta} &\mapsto \vec{\theta} = T(\vec{\zeta}). \end{aligned}$$

This transformation enables operations in the unconstrained space Θ while preserving a one-to-one correspondence with the original constrained space Λ. According to the change of variables theorem for probability densities, the joint density $p(\mathscr{D}, \vec{\zeta})$ needs to be rewritten as $p(\mathscr{D}, T^{-1}(\vec{\theta}))|\det J_{T^{-1}}(\vec{\theta})|$, where $J_{T^{-1}}$ is the Jacobian of T^{-1}. The ELBO is now expressed as:

$$\frac{1}{S}\sum_{s=1}^{S} \text{ELBO}(\vec{\theta}^{\,s}, \vec{\phi}) = \frac{1}{S}\sum_{s=1}^{S}\left[\ln \frac{p(\mathscr{D}, T^{-1}(\vec{\theta}^{\,s}))|\det J_{T^{-1}}(\vec{\theta})|}{q(\vec{\theta}^{\,s}|\vec{\phi})}\right], \tag{5.13}$$

(Kucukelbir et al., 2016).

All the up to here shown requirements allow for a straightforward implementation as TensorFlow Probability (Dillon et al., 2017) natively supports sampling by reparameterizing samples from auxiliary distributions, automatically computes the determinant of Jacobians associated with the change of variable transformations, and utilizes the JAX (Bradbury et al., 2018) autodifferentiation engine. The pseudocode for using SVI with automatic differentiation is shown in Algortihm 2. In this thesis the Adam optimizer is used for the SGD parameter updates (Kingma & Ba, 2017).

FCMN Parameterization: Since SGD operates in an unconstrained real space, the FCMN distribution requires reparameterization because covariance matrix estimations must be positive semi-definite, a property not guaranteed by unconstrained SGD optimization. This conflict can be circumvented by applying a log-Cholesky parameterization to the covariance matrix making it appropriate to use for gradient-based optimization methods while preserving the essential structural properties of the covariance matrix (Kucukelbir et al., 2016; Pinheiro & Bates, 1996).

Algorithm 2 SVI with Automatic Differentiation, Reparameterization (Kingma & Welling, 2013; Kucukelbir et al., 2016)

Input: A model $p(\mathscr{D}, \vec{\theta})$, the FCMN variational distribution $q(\vec{\theta} \mid \vec{\phi})$, a noise distribution $p(\vec{\epsilon})$, and a dataset $\mathscr{D}$
Output: Optimized parameters $\vec{\theta}$
$\vec{\phi} \leftarrow$ Initialize variational parameters
while the ELBO has not converged **do**
  $\vec{\epsilon} \leftarrow$ Random samples from noise distribution $p(\vec{\epsilon})$
  ELBO$(\vec{\theta}^{\,s}, \vec{\phi}) \leftarrow$ evaluate_elbo$(\mathscr{D}, T^{-1}(\vec{\theta}^{\,s}))$
  $\nabla_{\vec{\phi}}$ELBO$(\vec{\theta}^{\,s}, \vec{\phi}) \leftarrow$ automatic_differentiation(ELBO$(\vec{\theta}^{\,s}, \vec{\phi})$)
  $\vec{\phi} \leftarrow$ Update parameters using SGD($\nabla_{\vec{\phi}}$ELBO$(\vec{\theta}^{\,s}, \vec{\phi})$)
end while
return Optimal $\vec{\phi}$

In the log-Cholesky parameterization, a covariance matrix Σ is represented through its Cholesky factorization $\Sigma = LL^T$, where L is a lower triangular matrix. During optimization, the diagonal elements of L are log-transformed to maintain unconstrained optimization. When diagonal elements of L are restricted to positive values, L becomes unique. The parameter vector $\vec{L}_\phi = (\log(l_{11}), \log(l_{22}), \ldots, \log(l_{mm}), l_{21}, l_{31}, \ldots, l_{m,m-1})^T$ includes logarithms of the diagonal elements l_{kk} rather than the elements themselves, while the below-diagonal elements l_{kl} $(k > l)$ remain unchanged. While a full $m \times m$ covariance matrix requires $m(m+1)/2$ parameters due to symmetry, the Cholesky decomposition also needs $m(m+1)/2$ parameters: m for the diagonal and $m(m-1)/2$ for the below-diagonal elements. The log-Cholesky parameter vector $\vec{\theta}$ maintains the same number of parameters, ensuring that optimization proceeds in an unconstrained manner and that the resulting covariance matrix remains positive definite. Instead of the logarithm this thesis used the inverse Softplus transformation.

Consequently, in this thesis $\vec{\phi} = (\vec{\mu_\phi}, \vec{L}_\phi)$ and thus

$$q(\vec{\theta} \mid \vec{\phi}) = q(\vec{\theta} \mid (\vec{\mu_\phi}, \vec{L}_\phi)) = \mathrm{N}(\vec{\theta}\,; \vec{\mu}_\phi, L_\phi L_\phi^T), \tag{5.14}$$

with $\vec{\mu}_\phi$ having the same dimensionality as $\vec{\theta}$, i. e. $\vec{\mu}_\phi, \vec{\theta} \in \mathbb{R}^m$, and thus the dimensionality of the variational parameters' space Φ being $c = m + m(m+1)/2$.

5.2.1 SVI for SADR with a FCMN Variational Distribution

Defining the latent variables vector $\vec{\theta}$ when implementing the automatic differentiation SVI with the FCMN variational distribution approach for the GPD SADR model specified in Chapter 4.4 results in

$$\vec{\theta} = (\vec{\gamma}_\sigma, \vec{\gamma}_\xi, \beta_{0\sigma}, \beta_{0\xi}, \lambda_\sigma, \lambda_\xi)^T,$$

and for the ALD SADR model

$$\vec{\theta} = (\vec{\gamma}_{u_\tau}, \vec{\gamma}_{\psi}, \beta_{0u_\tau}, \beta_{0\psi}, \lambda_{u_\tau}, \lambda_{\psi})^T.$$

The bijective, differentiable function T^{-1} for transforming unconstrained latent variables to the constrained parameter space is the identity function for all parameters except the variance parameters λ^2. This applies because all parameters except λ^2 are defined on the real numbers, while λ^2 is defined on the positive real numbers. The chosen mapping is presented in Chapter 6.

Consequently, the ELBO for the SADR model with a FCMN variational distribution can be formulated explicitly. Let $T : \mathbb{R}_+ \to \mathbb{R}$ be a bijective transformation that maps constrained parameters $\lambda_k^2 \in \mathbb{R}_+$ with $k \in \{\sigma, \xi\}$ to the unconstrained space $\mathbb{R}$. Correspondingly, the inverse transformation $T^{-1} : \mathbb{R} \to \mathbb{R}_+$ maps values from the unconstrained space back to the constrained positive real space. Let $p(\vec{\epsilon})$ be defined as a Multivariate Normal (MN) distribution $\text{MN}(\vec{0}_m, I_m)$ and $g_{\vec{\phi}}(\vec{\epsilon}) = g_{\{\vec{\mu}_\phi, L_\phi\}}(\vec{\epsilon}) = \vec{\mu}_\phi + L_\phi \cdot \vec{\epsilon}$. This yields:

$$\text{ELBO}(\vec{\theta}, \vec{\phi}) = \mathbb{E}_{\vec{\epsilon} \sim p(\vec{\epsilon})} \left[\ln \frac{p(\mathscr{D}, \vec{\theta}^s)}{q(\vec{\theta}^s | \vec{\phi})} \right] \text{ with } \vec{\theta}^s = g_{\vec{\phi}}(\vec{\epsilon}^s) \tag{5.15}$$

$$\overset{\text{MC}}{\approx} \frac{1}{S} \sum_{s=1}^{S} \Bigg[\ln \text{GP}(\vec{y}^{\text{GP}} | \sigma^s(\vec{x}^{\text{GP}}), \xi^s(\vec{x}^{\text{GP}})) + \sum_{k \in \{\sigma, \xi\}} \left[\ln \text{DMN}\left(\vec{\gamma}_k^s | \vec{0}, \frac{\mathbf{K}^{\text{GP}}}{T^{-1}(\lambda_k^{2,s})} \right) \right. \tag{5.16}$$

$$\left. + \ln \text{N}(\beta_{0k}^s | \mu_k, \sigma_k) + \ln \text{IG}(T^{-1}(\lambda_k^{2,s}) | a_k, b_k) \right] - \ln \text{N}(\vec{\theta}^s | \vec{\mu}_\phi, L_\phi L_\phi^T) \Bigg]. \tag{5.17}$$

Within this framework, the ALD model can similarly be utilized by swapping out the GPD likelihood for the ALD likelihood and adjusting the latent variable vector. This modification possibility underlines how versatile the method is when dealing with different response distribution assumptions in the SADR setup.

Chapter 6

Implementation Details

This chapter aggregates implementation details such as the model hyperparameter settings (Section 6.1), the SVI and MCMC algorithms (Sections 6.2 and 6.3), the SVI and MCMC algorithms initialization (Section 6.4), the problem dimensionality for SVI and MCMC (Section 6.5), the modified GPD (Section 6.6), diagnostic tools including the Highest Density Interval (HDI) (Section 6.7) and the GPD-specific QQ plot (Section 6.9.5), the simulation study design and simulation study relevant implementations (Section 6.8), the case study design and case study relevant implementations (Section 6.9), and the computational environment (Section 6.10).

6.1 Model Hyperparameter Configuration

The hyperparameters for the model defined in Chapter 4 are mostly the same for both the simulation and the case study. The hyperparameters chosen showed the most promising convergence behavior for MCMC.

For SADR with a GP response, following Lang and Brezger (2004), the intercepts $\beta_{0\sigma}$ and $\beta_{0\xi}$ received weakly informative Normal priors, the variance parameters λ_σ^2 and λ_ξ^2 received weakly informative Inverse Gamma priors, and the regression coefficients $\vec{\gamma}_\sigma$ and $\vec{\gamma}_\xi$ received informative DMN priors with a location defined by a zero-vector, i. e.

$$\begin{aligned} \beta_{0\sigma} &\sim \mathrm{N}(0.0, 100.0) \quad \lambda_\sigma^2 \sim \mathrm{IG}(1.0, 0.005) \quad \vec{\gamma}_\sigma \sim \mathrm{DMN}(\vec{0}, \tfrac{1}{\lambda_\sigma^2} K) \\ \beta_{0\xi} &\sim \mathrm{N}(0.0, 10.0) \quad \lambda_\xi^2 \sim \mathrm{IG}(1.0, 0.005) \quad \vec{\gamma}_\xi \sim \mathrm{DMN}(\vec{0}, \tfrac{1}{\lambda_\xi^2} K). \end{aligned}$$

Analogous to SADR with an AL response, the hyperparameter configuration is the same except for $\beta_{0\xi}$

which has a wider prior, i. e.

$$\begin{aligned} \beta_{0\sigma} &\sim \mathrm{N}(0.0, 100.0) \quad \lambda_\sigma^2 \sim \mathrm{IG}(1.0, 0.005) \quad \vec{\gamma}_\sigma \sim \mathrm{DMN}(\vec{0}, \tfrac{1}{\lambda_\sigma^2} K) \\ \beta_{0\xi} &\sim \mathrm{N}(0.0, 100.0) \quad \lambda_\xi^2 \sim \mathrm{IG}(1.0, 0.005) \quad \vec{\gamma}_\xi \sim \mathrm{DMN}(\vec{0}, \tfrac{1}{\lambda_\xi^2} K). \end{aligned}$$

Regarding the P-splines, as suggested by Eilers and Marx (1996), throughout the whole thesis a moderately large number of 20 knots and a degree of 3 were used. The knot placement was always equidistant.

Given this model configuration, $\beta_{0\sigma}$, $\beta_{0\xi}$, λ_σ^2, and λ_ξ^2 are univariat-dimensional latent variables, while $\vec{\gamma}_\sigma$ and $\vec{\gamma}_\xi$ are 21-dimensional latent vectors.

6.2 SVI Algorithm

This thesis includes a Python implementation of SVI as detailed in Chapter 5 using the JAX (Bradbury et al., 2018) framework. For reproducibility purposes, this implementation—named VIGAMLSS—is structured like a package, although it is not formalized as an installable package. The code is accessible at https://github.com/marweda/master-s-thesis.

VIGAMLSS provides a modeling pipeline that includes the design matrix construction, the model specification, and the SVI posterior inference. The implementation methodology for SADR employed in both the simulation and case studies is presented in the following.

The inference pipeline commences with importing the necessary components, including an optimizer from the OPTAX package (DeepMind et al., 2020), the `DataPreparator` object, and the necessary distribution classes for the response, the priors, and the variational distribution:

```
import jax.numpy as jnp
from optax import adam
from vigamlss import (
    DataPreparator,
    DMN,
    IG,
    FCMN,
    ZeroCenteredGP,
    AL,
)
```

The `DataPreparator` class generates a design matrix. It requires the following arguments: the name assigned to the resulting `DesignMatrix` object; a one-dimensional covariate array `data`; the type of basis transformation; a boolean flag `intercept` determining if an intercept should be included; a `standardize` flag indicating whether the data should be standardized prior to a basis transformation; spline `degree`; the number of interior knots `num_knots`; the knot placement method `use_quantile`, which, if set to `False`, creates equidistant knots, and if set to `True`, creates quantile-based knots; and a boolean `return_knots` indicating whether the knot locations should be returned.

The class returns three outputs: a `DesignMatrix` object which holds the constructed design matrix and information about it, the penalty matrix K, and the array of knot positions used in the transformation.

```
DesignMatrix, K, knots = DataPreparator(
    name="DesignMatrix",
    data=X,
    basis_transformation="pspline",
    intercept=False,
    standardize=False,
    degree=3,
    num_knots=20,
    use_quantile=False,
    return_knots=True,
)()
```

To define the model, random variables within the model are assigned their respective distributions:

```
β0_scale = Normal("intercept_scale", jnp.array([0.0]), jnp.array([100.0]), size=1)

λ_scale = IG("lambda_smooth_scale", jnp.array([1.0]), jnp.array([0.005]), size=1)

γ_scale = DMN("spline_scale_coef", K, λ_scale)

β0_shape = Normal("intercept_shape", jnp.array([0.0]), jnp.array([10.0]), size=1)

λ_shape = IG("lambda_smooth_shape", jnp.array([1.0]), jnp.array([0.005]), size=1)

γ_shape = DMN("spline_shape_coef", K, λ_shape)

Y = ZeroCenteredGP(
    "y_GP",
    β0_scale + DesignMatrix @ γ_scale,
    β0_shape + DesignMatrix @ γ_shape,
    responses=Y_SYN,
)
```

The distribution classes in VIGAMLSS function as wrapper classes around TensorFlow Probability objects, carrying additional metadata for the construction and inference of the model. An exception is the **DMN** class, which encapsulates the **MultivariateNormalDegenerate** class of the package Liesel (Riebl et al., 2023). The **DMN** wrapper initializes a zero-vector of appropriate dimensionality for the location parameter by default. The construction of linear predictors is implemented in such a way that a mathematically literal formulation in the code is possible. Thus, VIGAMLSS allows the formulation of arbitrary linear predictors using addition and matrix multiplication operations. The **ZeroCenteredGP** represents a modified GPD with location parameter fixed at 0.0, conforming to the formally defined scale-shaped shape in Chapter 2. For Bayesian quantile regression, the response distribution class is replaced with the **AL** class.

The required link/response function application for SADR and the change of variables transformations necessary to map the variance parameters λ^2 onto the real number line are handled internally through automated processes. For link/response functions and for transformations, TensorFlow Probability `Bijector` classes are used, allowing automatic evaluation of link/response functions and their determinants of the Jacobian. For the $\mathbb{R} \to \mathbb{R}_+$ response function, the Softplus function is utilized (Wiemann, 2024). Similarly, Softplus is used for the $\mathbb{R} \to \mathbb{R}_+$ transformation of the λ^2 variance parameters that have been optimized in $\mathbb{R}$ by the SGD algorithm but are restricted to $\mathbb{R}_+$.

Once the model is created, the SVI optimization is performed by accessing the model as an attribute of `Y` and calling the `run_svi_optimization` method:

```
Y.model.run_svi_optimization(
    optimizer=adam,
    vi_dist=FCMNormal,
    vi_sample_size=64,
    epochs=epochs,
    mb_size=None,
    lr=0.001,
    max_norm=1.0,
    clip_min_max_enabled=True,
    zero_nans_enabled=True,
    prng_key=1,
    scheduler_type="constant",
)
```

The `run_svi_optimization` method accommodates any Optax optimizer class. Apart from a user-specified learning rate (lr) and scheduler type, the Adam optimizer (Kingma & Ba, 2017) is employed with default Optax configurations. The lr scheduler types used in this thesis include a constant scheduler and a scheduler that features a warm-up phase followed by a cosine decay. For the latter, additional keyword arguments (kwargs) must be passed to the `run_svi_optimization` method: `warmup_fraction` (denoting the proportion of total epochs allocated to the warm-up phase), `init_value` (specifying the initial learning rate at the start of the warm-up), and `end_value` (specifying the target learning rate for the final epoch to be achieved by cosine decay). The `vi_sample_size` parameter corresponds to S in Chapter 5. Mini-batching functionality is optional; specifying `None` results in the utilization of the complete dataset during each epoch. The parameters `max_norm`, `clip_min_max_enabled`, and `zero_nans_enabled` govern gradient operations described in Section 6.2.1.

The `vi_dist` parameter accepts a VIGAMLSS variational distribution class that encapsulates a TensorFlow Probability distribution class and initializes them. The only requirement for a TensorFlow Probability distribution to be used as a variational distribution is that its sampling method supports the reparameterization trick.

The initialization for `FCMN` is shown in Section 6.4, and the algorithm's arguments for the simulation and the case study are detailed in their respective sections.

6.2.1 Gradient Transformations

To ensure a defined gradient during each epoch, various gradient transformations can be applied sequentially before the main parameter update is performed. If enabled, the transformations are applied in the following order:

1. If `zero_nans_enabled` is `True`, the optimizer substitutes so-called not a number (nan) gradient values with zeros, preventing numerical instability by eliminating undefined values that would otherwise propagate through and potentially destabilize the optimization process.

2. If `max_norm` is not `None` but a float object, the optimizer scales the gradient values down when their global L2 norm exceeds the specified threshold `max_norm`, preserving the direction of the gradient vector while limiting its magnitude to mitigate gradient explosion (Pascanu et al., 2012).

3. If `clip_min_max_enabled` is `True`, the optimizer calls a function that replaces infinite gradient values with extreme finite values found within the same tensor. Negative infinity values are replaced with the minimum finite value, and positive infinity values are replaced with the maximum finite value, thereby ensuring defined gradients while preserving directional information (G. Callegher, personal communication, September 10, 2024).

6.3 MCMC Algorithm

Inference for the Bayesian SADR model with GP distributed responses via MCMC sampling methods was implemented using the Python package LIESEL (Riebl et al., 2023). The MCMC implementation follows the approach in Randell et al. (2016) for structured additive predictors for GP distributed responses using a combination of Gibbs and Metropolis-Hastings sampling. However, instead of Metropolis-Hastings sampling, this thesis utilized a Hamiltonian Monte Carlo sampler—specifically the No-U-Turn Sampler (NUTS) (Homan & Gelman, 2014). LIESEL uses a NUTS implementation fully written in JAX.

For the variance parameters λ_σ^2 and λ_ξ^2, Gibbs sampling is employed (Lang & Brezger, 2004; Randell et al., 2016), exploiting conditional independence properties inherent in the model to generate proposals with acceptance probability 1 (Geman and Geman, 1984; Murphy, 2023, p. 499; Randell et al., 2016).[1] The full conditionals of the variance parameters follow an Inverse Gamma distribution with updated parameters:

$$a' = a + \frac{\text{rank}(K)}{2}, \tag{6.1}$$

$$b' = b + \frac{1}{2}\gamma^\top K \gamma, \tag{6.2}$$

with a, a' and b, b' being the concentration and scale parameters, respectively (Lang & Brezger, 2004).

[1] Although Klein and Kneib (2016) and Gelman et al. (2014, p. 130) advise using Half-Cauchy distributions as weakly informative priors for variance parameters over an Inverse Gamma distribution, experiments in this thesis with Half-Cauchy priors yielded inferior MCMC convergence metrics. The Inverse Gamma prior with Gibbs sampling produced better $\hat{R}$ and ESS metrics compared to using NUTS with Half-Cauchy priors.

For the remaining parameters the NUTS sampler was used, i. e. for the global intercepts $\beta_{0\sigma}$ and $\beta_{0\xi}$ and the P-spline coefficients $\vec{\gamma}_\sigma$ and $\vec{\gamma}_\xi$ for scale and shape, respectively.

The parameter initialization for MCMC is presented in Section 6.4, and the algorithm's arguments for simulation and case study are detailed in their respective sections.

6.4 SVI & MCMC: Initialization

This section details the initialization strategies employed for both the SVI and MCMC inference approaches. Proper initialization is crucial to mitigate numerical instabilities during the early stages of optimization or sampling.

For the SVI implementation, the variational distribution `FCMN` is always initialized identically in all optimization runs. The location vector is initialized to zero, while the covariance structure is represented by a lower Cholesky decomposition triangle matrix with 0.01 along the diagonal and zeros in off-diagonal positions. This configuration corresponds to a covariance matrix with 0.0001 variances on the diagonal, yielding a narrow initial variational distribution that allowed for non-exploding and defined gradients at the beginning of the SGD procedure.

The MCMC implementation employs identical initialization values in both the simulation and case study. As shown in Table 6.1, the P-spline coefficients $\vec{\gamma}$ for the scale and shape parameters are initialized at zero, the scale intercept $\beta_{0\sigma}$ at 1.0, and the shape intercept $\beta_{0\xi}$ at 0.0. The variance parameters λ^2 for both the scale and shape components are initialized at 0.1. All MCMC experiments use a random seed of 1 for reproducibility.[2]

Table 6.1: MCMC Initialization Parameters

Parameter	Initial Value
P-spline coefficients $\vec{\gamma}$ (scale and shape)	0
Scale intercept $\beta_{0\sigma}$	1.0
Shape intercept $\beta_{0\xi}$	0.0
Variance parameters λ^2 (scale and shape)	0.1

6.5 SVI & MCMC: Inference Problem Dimensionality

Although both inference methods target the posterior distribution over the same latent variables, SVI and MCMC operate in spaces of considerably different dimensionality. The MCMC algorithm directly samples from the joint posterior distribution of the 46 model parameters, thus operating in a moderate-dimensional space. In contrast, as detailed in Chapter 5, SVI approximates the joint posterior by optimizing the variational distribution's variational parameters. The FCMN is parameterized by a location vector and

[2] Although it should be noted that full reproducibility with MCMC is limited due to the stochastic nature of the algorithm, see https://docs.liesel-project.org/en/latest/tutorials/md/05-reproducibility.html.

a covariance matrix represented through its Cholesky decomposition, i. e. for a m-dimensional model with $m = 46$ parameters, the variational distribution requires estimation of m location parameters plus the non-zero elements of the lower triangular Cholesky factor.

The number of non-zero elements in the lower triangular matrix of a $m \times m$ covariance matrix's Cholesky decomposition is given by:

$$\frac{m(m+1)}{2}.$$

Therefore, the SVI procedure must optimize a total of

$$46 + \frac{46(46+1)}{2} = 46 + 1081 = 1127$$

parameters.

6.6 Modified Generalized Pareto Distribution

The log PDF of the GPD is required for computing the ELBO in SVI optimization as to be seen in Chapter 5. The GPD's parameter-dependent support (see Chapter 2) presents a computational challenge for the SVI optimization process. For responses outside the support, the naïve log PDF results in $-\infty$, causing undefined gradients that impede optimization. For the GEV, which also has parameter-dependent support, this problem is addressed by J. Brachem (personal communication, April 01, 2025) with a modified log PDF that returns finite values for OOS responses, implicitly penalizing the parameter values that caused the OOS cases. The same modification was applied to the GPD by G. Callegher (personal communication, 10 September 2024).

Let $(Y_i)_{i=1}^N$ be a sequence of independent, but not necessarily identical, continuous random variables. For each i, let y_i denote the realization of Y_i, i. e., the observed response. Given a GPD with parameters $\mu \in \mathbb{R}$ (location), $\sigma \in \mathbb{R}_+$ (scale), and $\xi \in \mathbb{R}$ (shape), the support $\mathscr{X}$ is defined as:

$$\mathscr{X} = \begin{cases} \{y_i \in \mathbb{R} : y_i \geq \mu\}, & \text{if } \xi \geq 0, \\ \{y_i \in \mathbb{R} : \mu \leq y_i \leq \mu - \sigma/\xi\}, & \text{if } \xi < 0, \end{cases} \tag{6.3}$$

(Embrechts et al., 1997, p. 162–164).

Let $\ell(y_i; \mu, \sigma, \xi)$ denote the naïve log PDF of the GPD. The modified log-PDF $\tilde{\ell}(y_i; \mu, \sigma, \xi, \nu)$, with penalty parameter $\nu > 0$, is defined as:

$$\tilde{\ell}(y_i; \mu, \sigma, \xi, \nu) = \begin{cases} \ell(y_i; \mu, \sigma, \xi), & \text{if } y_i \in \mathscr{X}, \\ (\alpha(y_i; \mu, \sigma, \xi) - 0.5) \cdot \frac{\nu}{m}, & \text{if } y_i \notin \mathscr{X}, \end{cases} \tag{6.4}$$

where $m = \max(n_{\text{violations}}, 1)$ with $n_{\text{violations}}$ is the count of responses that violate support constraints which scales the penalty per violating sample, and $\alpha(y_i; \mu, \sigma, \xi)$ is the signed distance to the support

boundary:

$$\alpha(y_i;\mu,\sigma,\xi) = \begin{cases} y_i - \mu, & \text{if } y_i \le \mu, \\ 1 + \xi \cdot \frac{y_i - \mu}{\sigma}, & \text{if } \xi < 0 \text{ and } y_i > \mu. \end{cases} \tag{6.5}$$

The function $\alpha(y_i;\mu,\sigma,\xi)$ encodes the signed distance to the relevant support boundary of the GPD as follows:

- **Case 1:** $y_i \le \mu$

$$\alpha(y_i;\mu,\sigma,\xi) = y_i - \mu \tag{6.6}$$

  This function measures the distance from y_i to the lower support boundary which is maintained for any ξ, with:

  - At the lower boundary: $z = 0$ when $y_i = \mu$
  - Beyond the lower boundary: $z < 0$ when $y_i < \mu$ (increasingly negative)

  The magnitude of z increases as y_i moves farther from the boundary.

- **Case 2:** $\xi < 0$ **and** $y_i > \mu$

$$\alpha(y_i;\mu,\sigma,\xi) = 1 + \xi \cdot \frac{y_i - \mu}{\sigma} \tag{6.7}$$

  When $\xi < 0$, the support has both lower (μ) and upper ($\mu - \sigma/\xi$) boundaries. For points above μ, this function measures the relative position with respect to the upper boundary:

  - At the upper boundary ($y_i = \mu - \sigma/\xi$): $z = 0$
  - Beyond the upper boundary ($y_i > \mu - \sigma/\xi$): $z < 0$ (increasingly negative)

In both cases, z provides a continuous measure that is zero at the boundary and negative outside the support, with the magnitude indicating the distance from the relevant boundary. To avoid distances of 0 or close to zero, which could lead to numerical instabilities, z is subtracted by 0.5.

The modified log-PDF $\tilde{\ell}$ affects the negative ELBO in two ways:

- **For in-support values** ($y_i \in \mathcal{X}$): $\tilde{\ell} = \ell$, i. e. the naïve GPD log-PDF contributes to the ELBO. Higher log-densities in these regions reduce the negative ELBO, indicating a better model fit.
- **For out-of-support values** ($y_i \notin \mathcal{X}$): The term $(\alpha(y_i) - 0.5) \cdot \frac{\nu}{m}$ applies:
  - When $y_i \le \mu$: $\alpha(y_i) = y_i - \mu < 0$, resulting in a term $\propto -\frac{\nu}{m}$
  - When $\xi < 0$ and $y_i > \mu - \sigma/\xi$: $\alpha(y_i) = 1 + \xi \cdot \frac{y_i - \mu}{\sigma} < 0$, again resulting in a term $\propto -\frac{\nu}{m}$

These possibly large negative terms make up an improper log PDF value, which increases the negative ELBO, creating a gradient that directs the variational parameters to the support region $\mathscr{X}$ which includes all responses and thus acts like a penalty.

6.7 Highest Density Intervals

In Bayesian inference, the HDI is a method for constructing credible intervals that include the most likely parameter values. In this thesis they were used to depict the uncertainty. An HDI interval is defined by including all parameter values whose posterior density minimizes the interval width while maintaining the required probability coverage (Carlin & Louis, 2009, pp. 48–49). This thesis uses the package ArviZ (Kumar et al., 2019) and its function `arviz.hdi` to compute the HDI utilizing posterior samples.

6.8 Simulation Study

In the context of the SADR framework with GPD-distributed responses, the simulation study evaluates the posterior inference capabilities of SVI using a modified GPD by comparing the WD to a MCMC baseline using an naïve GPD across various sample sizes. A single MCMC run establishes a baseline posterior for each sample size, which is then compared using the WD against inferred posteriors from one hundred SVI and MCMC runs. The one hundred MCMC runs serve as a reference because WD lacks an absolute scale (Callegher et al., 2025). One hundred SVI runs, one hundred MCMC runs, and additional single SVI runs served to also assess the out-of-sample (OOS) behavior of both inference methods using the modified and naïve GPD, respectively. The simulation study is based on simulated data, introduced in Section 6.8.1. Furthermore, Section 6.8.2 is about the chosen configuration for the additional single and the one hundred SVI and MCMC run. Section 6.8.5 motivates the Sinkhorn-based Wasserstein distance approximation. All runs conform to the model, algorithm, and initialization configurations detailed in Sections 6.1–6.4. In summary, for each N

- a single run MCMC establishes the baseline inferred posterior to be compared to;
- the comparison includes 100 reference-MCMC runs against 100 SVI runs at 1000, 10000, 20000, and 50000 epochs, with performance evaluated using WD and SD metrics against the MCMC baseline;
- the 100 MCMC and SVI runs are also used to analyze the OOS behaviour;
- a 20000 epochs single SVI run for each of the three sample sizes is retained to track OOS violations over the course of the optimization;
- a 20000 epochs single-run SVI is retained to relate the modifed GPD's behaviour w. r. t. OOS cases to the ELBO trajectory.

6.8.1 Simulated Data

The simulated data were generated following a GPD with covariate-dependent scale and shape parameters (see Figure 6.1 for 1000 independently simulated data points). The location parameter was consistently set to zero and is therefore left out in the subsequent notation. A univariate covariate x_i was drawn from a uniform distribution on $[-3,3]$ and sorted in ascending order: $x_i \sim U(-3,3)$. A response y_i was then generated from GP distributions with covariate-dependent parameters: $y_i \sim \text{GP}(\sigma(x_i), \xi(x_i))$, where the scale and shape parameter functions were defined as:

$$\sigma(x_i) = \begin{cases} 0.1(x_i - 1)^2 + 1.5, & \text{if } x_i < 1 \\ 1.5, & \text{otherwise,} \end{cases}$$

and

$$\xi(x_i) = \begin{cases} -(0.12\tanh(x_i) + 0.01x_i^2 - 0.1), & \text{if } x_i < 1 \\ -(x_i - 1)0.1, & \text{otherwise.} \end{cases}$$

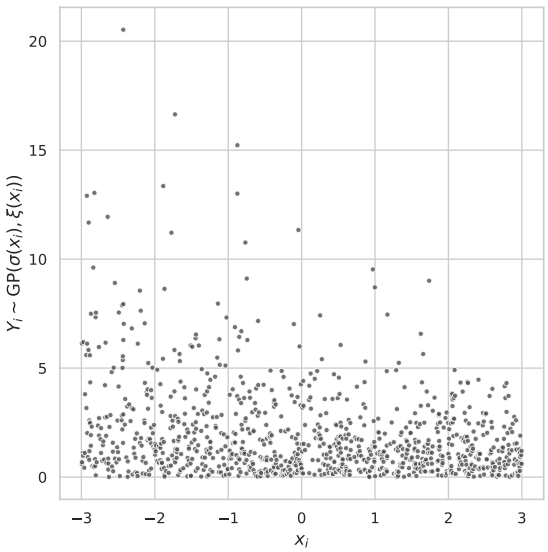


Figure 6.1: Simulated Data for Regression with GPD Responses: $N = 1000$

These functions were selected to create non-linear relationships between the covariate and the GPD parameters (see Figure 6.2).

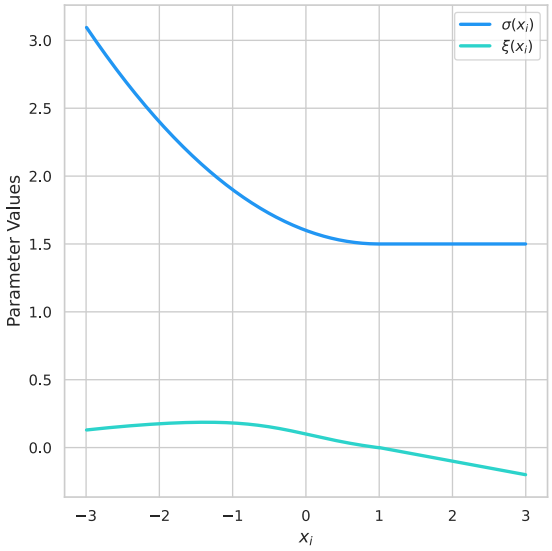


Figure 6.2: Simulated Data's True Parameter Variation w. r. t. x

6.8.2 SVI & MCMC Configuration

The simulation study utilizes one hundred runs and some additional single runs with different configurations to evaluate the performance of the SVI and MCMC methods. Table 6.2 presents the SVI configuration for additional single runs and one hundred executions, respectively. The NUTS sampler used the default LIESEL parameter settings, except for the `da_target_accept`, `max_treedepth`, and `initial_step_size`—Table 6.3 provides the configuration for the baseline single runs and for the one hundred executions.[3,4]

6.8.3 SVI Loop

The implementation of the SVI simulation study loop follows Algorithm 3, which systematically captures SVI inference results across various parameter configurations set in 6.8.2.

The algorithm processes four epoch counts 1000, 10000, 20000, and 50000, iterating through three sample sizes 250, 500, and 1000. For each sample size, it conducts 100 runs initialized with unique random seeds derived from the run index and sample size. Each iteration simulates data, prepares the model inputs, defines the model structure, and executes SVI optimization. The procedure captures the optimized parameters after convergence, enabling comparative analysis of SVI performance under varying conditions

[3]Following Singmann (2016), this thesis chose these high values for the `da_target_accept`, `initial_step_size`, and the `max_treedepth`, because the model exhibited divergent transitions in the post warm-up posterior sampling phase for lower values and where only increasing the `da_target_accept` wasn't sufficient enough to fix the divergent transitions issues. Divergent transitions in Hamiltonian Monte Carlo compromises the estimator's validity, as divergences despite a `da_target_accept` increase indicate severe model pathologies and significant bias in the resulting estimators (Betancourt, 2018).

[4]The number of warm-up iterations (Murphy, 2023, pp. 514–515) was selected to ensure satisfactory convergence behavior. Posterior sample quantity was determined to achieve an adequate effective sample size (Murphy, 2023, pp. 523–524).

Table 6.2: SVI Simulation Study Configuration Arguments: Loop and Single-run

Parameters	Loop	Single-run
Y	ZeroCenteredGPD	
optimizer	adam	
vi_dist	FCMNormal	
vi_sample_size	64	
epochs	[1000,10000,20000,50000]	20000
mb_size	None	
lr	1e-3	
scheduler	"constant"	
scheduler kwargs	None	
max_norm	1.0	
clip_min_max_enabled	True	
zero_nans_enabled	True	
prng_key	See Section 6.8.3	1

Table 6.3: MCMC Simulation Study Configuration Arguments: Loop and Baseline-runs

Parameter	Loop	Baseline-runs
NUTS da_target_accept	0.999 (N=250, N=500, N=1000)	
NUTS max_treedepth	20	
NUTS initial_step_size	0.001	
Warm-up iterations	12000	
Posterior samples per chain	6000	
Random seed	See Section 6.8.4	1

Algorithm 3 Simulation Study Loop for SVI (Callegher et al., 2025)

Input: Number of runs n_runs = 100, sample sizes Ns = [250, 500, 1000],
epochs list epochs_list = [1000, 10000, 20000, 50000]
Output: Optimization results for all parameter combinations
for each epochs in epochs_list **do**
  for each sample size N in Ns **do**
    for run_i = 0 to n_runs − 1 **do**
      Generate random seed based on run_i and N
      Split seed into two separate seeds
      Generate synthetic data using first seed
      Prepare data for inference
      Define model using prepared data
      Run optimization with specified epochs using second seed
      Collect optimization results
    end for
  end for
end for
return Results

and optimization parameters.

6.8.4 MCMC Loop

The implementation of the MCMC simulation study follows Algorithm 4, which systematically captures MCMC inference results across various parameter configurations set in 6.8.2.

Algorithm 4 Simulation Study Loop for MCMC (Callegher et al., 2025)

```
Input: Number of runs n_runs = 100, sample sizes Ns = [250, 500, 1000]
Output: MCMC sampling results for all parameter combinations
for each sample size N in Ns do
    Create directory for results with current N
    for run_i = 0 to n_runs − 1 do
        Generate random seed based on run_i and N
        Split seed into two separate seeds
        Generate synthetic data using first seed
        Prepare data for inference
        Run MCMC sampling using the second seed
        Collect sampling results
        Store results
    end for
end for
return Results
```

The algorithm is analogues to Algorithm 3, but MCMC-specific. It processes the same three sample sizes 250, 500, and 1000. Each iteration generates synthetic data using the first seed, prepares the model inputs, and executes Markov Chain Monte Carlo sampling using the second seed.

6.8.5 Wasserstein Distance Approximation

The Wasserstein distance defines a metric between two probability distributions, providing a means to compute the dissimilarity between them using a set of samples from each distribution. Interpreting the dissimilarity between two probability distributions through the Wasserstein distance relies on its metric properties: (i) symmetry (ordering of distributions doesn't change the distance), (ii) strict positivity (distance is positive for different distributions), (iii) zero distance only for identical distributions, and (iv) the triangle inequality (distance between two distributions is not greater than the sum of distances through another distribution) (Peyré & Cuturi, 2020, p. 19).

As shown by Cuturi, for one-dimensional probability distributions represented by n Monte Carlo draws, the Wasserstein distance can be solved in $O(n \log n)$ operations. However, this efficiency does not extend to multidimensional cases. For probability distributions in $\mathbb{R}^d$ with $d \geq 2$, the computational complexity increases to at least $O(n^3 \log n)$. Although approximation techniques can achieve linear time complexity for dimensions $2 \leq d \leq 4$, these approaches become prohibitively expensive when d exceeds 4 due to exponential cost increases. This computational barrier severely limits the practical application of exact

Wasserstein distance calculations when the sample size n reaches a few thousand and when working with higher-dimensional probability distributions (Cuturi, 2013).

Cuturi (2013) introduces the Sinkhorn distance (SD), whose computation includes a regularization factor $1/\vartheta$[5], with which SD converges to the exact WD. Given the computationally unfeasible task of evaluating the Wasserstein distance for the marginal posteriors over the 21-dimensional random vectors $\vec{\gamma}_\sigma$ and $\vec{\gamma}_\xi$, and the 46-dimensional joint posterior, this thesis instead calculates the SD for them. This thesis computes the Sinkhorn distance using the package OTT-JAX (Cuturi et al., 2022). In the following, the package's notation[6] for the regularization factor $\omega = 1/\vartheta$ was used, meaning that the approximation becomes more accurate for $\omega \to 0$. Higher ω values result in faster convergence but larger approximation errors, while lower ω values produce sparser plans that more closely approximate the true optimal transport solution but require more iterations and risk numerical instability. In Cuturi (2013), an $\omega = 0.02$ and lower are considered to result in a high accuracy approximation where $\omega = 0.01$ in their application achieved a median relative gap between the true WD and approximated WD of 1.2%. The regularization parameter of choice in this thesis was $\omega = 0.01$.

For the one-dimensional marginal posteriors over $\beta_{0\sigma}$, $\beta_{0\xi}$, λ_σ^2, and λ_ξ^2 the Wasserstein distance was computed using the package POT (Flamary et al., 2021).

The default parameter configurations were used in OTT-JAX and POT.

6.8.6 Thinning Samples

Thinning refers to the practice of retaining only every k-th draw from a Markov chain, discarding intermediate samples. In complex posterior distributions where draws exhibit positive autocorrelation, thinning can reduce the correlation structure of the resulting chain. The autocorrelation at lag k in the thinned sequence corresponds to the autocorrelation at lag $k \cdot t$ in the original sequence, where t is the thinning interval. Although thinning generally reduces ESS compared to using all available draws, thinned samples typically achieve higher ESS than not thinned samples of equivalent size due to decreased autocorrelation between chains for longer chains. Thinning should generally be avoided when draws exhibit negative autocorrelation, as thinning in such cases would increase correlation and further reduce the ESS. The primary justification for thinning in practical applications is reducing memory requirements rather than improving statistical efficiency. This consideration becomes particularly relevant when processing large posterior samples for computationally intensive operations (Stan Development Team, 2024).

Thinning became necessary in this thesis because the available VRAM (see Section 6.10) posed a computational resource restriction, making it impossible to retain all posterior samples for multidimensional SD calculations using the GPU. Using the CPU was computationally too time-expensive.

[5] Cuturi (2013) uses the $1/\lambda$ for the regularization factor, but in order to avoid confusion with in this thesis already defined variance parameter λ^2, ϑ is used instead.

[6] Cuturi et al. (2022) uses the ϵ for the regularization factor, but in order to avoid confusion with in this thesis already defined auxiliary random variable ϵ, ω is used instead.

A thinning interval of 3 was applied to the posterior samples of the MCMC chains.

6.9 Case Study

The case study applies the two-stage PoT procedure to the *Danish Fire Insurance* data. The dataset is described in Section 6.9.1. Stage 1 uses SVI to fit an ALD for inferring the quantile-specific threshold and determine the exceedances (see Sections 6.9.3 and 6.9.4) and stage 2 fits the GPD to the exceedances (see Section 6.9.2). The GPD model fit is assessed with GP-specific QQ plots (Section 6.9.5). Both stages rely on the model and initialization setup of Sections 6.1–6.4. Regarding the second stage, SVI using the modified GPD is compared against MCMC using the naïve GPD. The stage 2 posterior inference with SVI and MCMC both rely on the same stage 1 SVI inferred threshold. The SVI and MCMC inferred posteriors were compared using kernel density estimates and the estimated GP scale and shape parameters. The kernel density estimates were computed using the SEABORN package's `seaborn.kdeplot` function (Waskom, 2021).

6.9.1 Data

The case study utilizes the *Danish Fire Insurance* dataset, which contains 2167 industrial fire insurance claims in Denmark from January 3, 1980, to December 31, 1990, with claim values measured in millions of Danish Krone (DKK). The dataset was obtained from the EVIR R package (Pfaff et al., 2018), where the original data was provided by *Mette Rytgaard of Copenhagen Re*. For analytical purposes, the preparation process involved extracting the temporal information from the 'times' attribute, creating a R dataframe with both loss values and corresponding dates, sorting chronologically, and calculating the elapsed time (in days) for a given record since the first recorded incident. The resulting dataset structure defines the response variable 'Total_loss' (representing the claim values in millions of DKK) as a function of the covariate 'Days' (representing the number of days elapsed since January 3, 1980, with this initial date counted as day zero).

The N losses are assumed to be realizations from a sequence of independent but not necessarily identical random variables $\{Y\}_i^N$ with y_i being a realization of Y_i and the i-th observed loss, i. e. the i-th response. The 'Days' covariate, represented by x_i, corresponds to the i-th response's covariate. For the first stage the N random variables are assumed to follow an ALD.

Given Table 6.4, the *Danish Fire Insurance* Dataset exhibits a heavily right-skewed distribution of total losses. The mean value of 3.39 million DKK substantially exceeds the median of 1.78 million DKK, indicating the presence of extreme losses. The interquartile range (1.65 million DKK) is notably compressed relative to the full range of 1.00 to 263.25 million DKK. The substantial gap between the 75^{th} percentile (2.97 million DKK) and the maximum value (263.25 million DKK) further confirms the distribution's heavy right tail. This pattern shows that most responses cluster at lower values while a small number of extreme events seem to exert disproportionate influence on the mean.

Table 6.4: Descriptive Statistics of the *Danish Fire Insurance* Dataset (1980-1990)

Variable	Measure	Value
Time Period	Calendar dates	January 3, 1980–December 31, 1990
	Days	0–4015
Total Loss (million DKK)	25^{th} percentile	1.32
	Median	1.78
	75^{th} percentile	2.97
	Mean	3.39
	Range	1.00–263.25

6.9.2 SVI & MCMC Configuration

The case study implemented the two-stage PoT approach (see Chapter 2). The first stage involved fitting an ALD to determine appropriate thresholds and threshold exceedances, while the second stage focused on fitting the GPD to the threshold exceedances. Table 6.5 shows the SVI stage 1 and stage 2 SVI configuration. The NUTS sampler used the default LIESEL parameter settings, except for the `da_target_accept`, maximal tree depth, and `initial_step_size`—Table 6.6 shows the chosen MCMC configuration.[7,8]

Table 6.5: Algorithm Configuration for Case Study: Stage 1 and Stage 2

Parameter	Stage 1: ALD	Stage 2: GPD
`optimizer`	`adam`	
`lr`	`1e-2`	`1e-3`
`scheduler`	`"warmup_cosine_decay"`	`"constant"`

`scheduler kwargs`	`warmup_fraction=0.05`
	`init_value=1e-5`
	`end_value=1e-4`
		`None`

`vi_dist`	`FCMNormal`	
`vi_sample_size`	`64`	
`mb_size`	`None`	
`max_norm`	`1.0`	
`clip_min_max_enabled`	`False`	`True`
`zero_nans_enabled`	`False`	`True`
`epochs`	`20000`	
`prng_key`	`1`	`1`

[7]Similarly to the simulation study, following Singmann (2016), this thesis chose these high values for the `da_target_accept`, `initial_step_size`, and the `max_treedepth`, because the model exhibited divergent transitions for lower values and where only increasing the `da_target_accept` wasn't sufficient enough to fix the divergent transitions issues. Divergent transitions in Hamiltonian Monte Carlo compromises the estimator's validity, as divergences despite a `da_target_accept` increase indicate severe model pathologies and significant bias in the resulting estimators (Betancourt, 2018).

[8]The number of warm-up iterations was selected to ensure satisfactory convergence behavior (Murphy, 2023, pp. 514–515). Posterior sample quantity was determined to achieve an adequate effective sample size (Murphy, 2023, pp. 523–524).

Table 6.6: MCMC Simulation Study Configuration Arguments

Parameter	Arguments
NUTS `da_target_acceptance`	`0.99`
NUTS `max_treedepth`	`20`
NUTS `initial_step_size`	`0.001`
Warm-up iterations	`6000`
Posterior samples per chain	`12000`
Random seed	`1`

6.9.3 Asymmetric Laplace Distribution: Implementation

The ALD is central to the Bayesian two-stage PoT approach as it underpins the Bayesian quantile regression used for threshold determination (see Chapter 3) (Youngman, 2019; Yu & Moyeed, 2001). Fitting the ALD using SVI requires evaluating its log PDF and generating samples from an ALD distributed random variable (see Chapter 5). Because TensorFlow Probability does not natively support the ALD, a custom distribution class was implemented for the three-parameter ALD parameterization that is used for Bayesian quantile regression (Yu & Moyeed, 2001; Yu & Zhang, 2005).

The three-parameter ALD, denoted as $\mathrm{AL}(y|u_\tau, \sigma)$ in this thesis (see Chapter 3), is characterized by a location parameter $u_\tau \in \mathbb{R}_+$ (corresponding to the τ-th quantile-level), a scale parameter $\sigma \in \mathbb{R}_+$, and a skewness parameter $\tau \in (0,1)$ (representing the quantile-level). The logarithm of the ALD's PDF (see (3.1)) was implemented as:

$$\log f(y;\mu,\sigma,\tau) = \log(\tau(1-\tau)) - \log(\sigma) - \vartheta_\tau\left(\frac{x-\mu}{\sigma}\right). \tag{6.8}$$

For drawing samples from an ALD distributed random variable, let υ and ι be independent random variables each following an exponential distribution with rate 1, i. e. $\upsilon \sim \mathrm{Exp}(1)$ and $\iota \sim \mathrm{Exp}(1)$. According to Yu and Zhang, 2005, the linear combination

$$X = \frac{\upsilon}{\tau} - \frac{\iota}{1-\tau} \tag{6.9}$$

is referred to as a standard ALD random variable, i. e. A follows the distribution $\mathrm{AL}(y|0,1,\tau)$. Subsequently, any $\mathrm{AL}(y|\mu,\sigma,\tau)$ random variable Y can be obtained through the affine map

$$Y = \mu + \sigma X, \tag{6.10}$$

Yu and Zhang, 2005.

6.9.4 Asymmetric Laplace Distribution: Threshold Selection

The selection of an appropriate threshold $u_\tau(\vec{x})$ for extreme value modeling using the PoT approach presents a fundamental challenge. As noted by Coles (2001, p. 78):

> [...] too low a threshold is likely to violate the asymptotic basis of the model, leading to bias; too high a threshold will generate few excesses with which the model can be estimated, leading to high variance. The standard practice is to adopt as low a threshold as possible, subject to the limit model providing a reasonable approximation.

Following Youngman (2019), a quantile regression approach was employed to define the threshold $u_\tau(\vec{x})$ corresponding to a fixed quantile-level τ, and after examining various quantiles through QQ diagnostic plots (see Section 6.9.5) as recommended, the quantile-level $\tau = 0.91$ was selected. This value provided suitable agreement in QQ plots and retained a sufficient number of exceedances (204 excesses) to ensure stable parameter estimation with reduced model variance.

6.9.5 Generalized Pareto Quantile-Quantile Plots

To assess the goodness-of-fit of the GPD models, QQ plots based on the probability integral transformation approach are used as detailed in Coles (2001, pp. 110–111). For each fitted model, both the observed threshold excesses $\vec{y}^{\,\mathrm{GP}}$ and theoretical quantiles are transformed to a common exponential scale, enabling the QQ plot diagnostic procedure for non-stationary parameters.

Let $\vec{y}^{\,\mathrm{GP}} = (y_\mathfrak{l} - u_\tau(x_\mathfrak{l}))_{i\in I}$ with $I = \{i : y_i > u_\tau(x_i), i = 1, 2, \ldots, N\}$ represent the vector of all y_i elements exceeding $u_\tau(x_i)$, centered by subtracting $u_\tau(x_\mathfrak{l})$ from each exceedance, with $|\vec{y}^{\,\mathrm{GP}}| = M$ as the number of exceedances. For a set of ordered threshold exceedances $\vec{y}^{\,\mathrm{GP}} = (y_1^{\mathrm{GP}} \le y_2^{\mathrm{GP}} \le \ldots \le y_M^{\mathrm{GP}})^T$, each exceedance $\mathfrak{l}$ has a corresponding estimated scale parameter $\tilde{\sigma}_\mathfrak{l}(x_\mathfrak{l})$ and shape parameter $\tilde{\xi}_\mathfrak{l}(x_\mathfrak{l})$. Here, $x_\mathfrak{l}$ is an element of vector $\vec{x}^{\,\mathrm{GP}}$, where $\vec{x}^{\,\mathrm{GP}} = (x_\mathfrak{l})_{\mathfrak{l}\in I}$ with the same index set I denotes the vector of all x_i for the corresponding response $y_i > u_\tau(x_i)$ that exceeds the threshold. The transformation to a standardized exponential scale is defined as:

$$\widetilde{y}_\mathfrak{l} = \begin{cases} \frac{1}{\tilde{\xi}_\mathfrak{l}(x_\mathfrak{l})} \log\left\{1 + \tilde{\xi}_\mathfrak{l}(x_\mathfrak{l})\left(\frac{y_\mathfrak{l}}{\tilde{\sigma}_\mathfrak{l}(x_\mathfrak{l})}\right)\right\} & \text{for } \tilde{\xi}_\mathfrak{l}(x_\mathfrak{l}) \neq 0, \\ \frac{y_\mathfrak{l}}{\tilde{\sigma}_\mathfrak{l}(x_\mathfrak{l})} & \text{for } \tilde{\xi}_\mathfrak{l}(x_\mathfrak{l}) \approx 0. \end{cases} \tag{6.11}$$

This transformation converts the GP distributed excesses $\vec{y}^{\,\mathrm{GP}}$ to variables that follow the standard exponential distribution. Within the implementation, the case $\tilde{\xi}_\mathfrak{l}(x_\mathfrak{l}) \approx 0$ utilizes the limiting form of the transformation if $|\tilde{\xi}_\mathfrak{l}(x_\mathfrak{l})|$ falls below the threshold of 1×10^{-8}.

For the theoretical quantiles, standard exponential quantiles corresponding to the empirical probabilities

are employed. Specifically, the theoretical quantiles are computed as:

$$z_{\mathfrak{l}} = -\log\left(1 - \frac{\mathfrak{l}}{M+1}\right), \quad \mathfrak{l} = 1, \ldots, M \tag{6.12}$$

where $\frac{\mathfrak{l}}{M+1}$ represents the empirical probability corresponding to the statistic of $\mathfrak{l}$-th order. The QQ plot consists of the paired points $\{(\widetilde{y}_{\mathfrak{l}}, z_{\mathfrak{l}}); \mathfrak{l} = 1, \ldots, M\}$. Under a correct model specification, these points should approximately follow a straight line with unit slope through the origin. Substantial deviations from this line indicate potential model inadequacies.

6.10 System Details

The computations were performed on a DELL XPS 17 9700 notebook with an INTEL CORE I7-10875H processor (2.30GHz, 16 threads) and 64GB RAM. The system utilizes an NVIDIA GEFORCE RTX 2060 MAX-Q GPU with 6GB VRAM, operating on UBUNTU 22.04.5 LTS (64-bit). The software implementation was performed using PYTHON 3.11.9 with the following package versions: SEABORN 0.13.2, JAX 0.5.0, LIESEL 0.3.3, TENSORFLOW PROBABILITY 0.24.0, CUDA 12.5.39, OPTAX 0.2.4, ARVIZ 0.18.0, POT 0.9.5, and OTT-JAX 0.5.0.

Chapter 7

Results

This chapter presents findings on the applicability of SVI for SADR under GP distributed responses from both a simulation and a case study.

7.1 Simulation Study

In the simulation study, simulated GP distributed responses with covariate-dependent scale and shape parameters were generated for sample sizes $N \in \{250, 500, 1000\}$. A Bayesian SADR model with GP distributed responses was assumed. The simulation study compared joint posterior distributions over scale SAP and shape SAP latent variables, along with their respective marginal posteriors, obtained through SVI with a modified GPD against those generated via MCMC with a naïve GPD. For each N

- a single run MCMC established the baseline inferred posterior to be compared to;
- the comparison included 100 reference-MCMC runs against 100 SVI runs at 1000, 10000, 20000, and 50000 epochs, with performance evaluated using WD and SD metrics against the MCMC baseline;
- the 100 MCMC and SVI runs were also used to analyze the OOS behaviour;
- a 20000 epochs single SVI run for each of the three sample sizes was retained to track OOS violations over the course of the optimization;
- a 20000 epochs single-run SVI was retained to relate the modifed GPD's behaviour w. r. t. OOS cases to the ELBO trajectory.

All model, algorithm configurations, and computational resources are documented in Chapter 6.

MCMC Diagnostics: The MCMC baselines achieved good convergence and mixing (see Table 7.1 and Appendix D.1.).

Metric	$N = 250$	$N = 500$	$N = 1000$
Worst $\hat{R}$	ca. 1.00432	ca. 1.00227	ca. 1.00135
ESS BULK (min–max)	Ca. 1196–6223	Ca. 1931–10166	Ca. 2516–11786
ESS TAIL (min–max)	Ca. 1494–7147	Ca. 3601–14407	Ca. 4436–18352
Funnel of Hell[1]	No	No	No

Table 7.1: Simulation Study Baseline MCMC Diagnostic Statistics for Various Sample Sizes
The effective sample size values refer to the non-thinned posterior samples.

The effective sample size values in Table 7.1 correspond to posterior samples that have not been thinned. As described in Section 6.8.6, thinning reduces the effective sample size. However, the thinned posterior samples still exhibit a higher effective sample size than non-thinned posterior samples of the same size. The thinning degree of 3 utilized in this analysis corresponds to 4000 posterior samples per chain. For context, Table 7.2 presents the effective sample sizes for the baseline MCMC runs achieved with 4000 posterior samples per chain.

Metric	$N = 250$	$N = 500$	$N = 1000$
ESS BULK (min–max)	Ca. 372–1938	Ca. 625–3923	Ca. 842–3771
ESS TAIL (min–max)	Ca. 597–2969	Ca. 1080–5070	Ca. 1400–5974

Table 7.2: Simulation Study Baseline MCMC Effective Sample Size References for thinned Posterior Samples
Table 7.2 serves as a reference point for evaluating thinning effects on effective sample sizes. The displayed baseline MCMC runs configurations differ only in the number of posterior samples per chain, fixed at 4000.

WD and SD comparison: Figure 7.1 and Table 7.3 indicate convergence across SVI epochs with negligible improvements observed beyond epoch 20000. All subsequent SVI-MCMC comparisons refer to SVI results at epoch 50000.

SVI demonstrates adequate approximation quality to MCMC in terms of SD for the joint distribution. At sample sizes $N = 250$ and $N = 500$, SVI exhibits SD values proximate to MCMC reference metrics, while performance marginally degrades at $N = 1000$.

SVI performance regarding SADR latent variables' marginal posteriors varies by variable type. Scale-related latent variables display the most persistent discrepancies. The SAP regression coefficient latent variables on scales $\vec{\gamma}_\sigma$ exhibit modest SD discrepancies that remain across sample sizes. The scale intercept $\beta_{0\sigma}$ demonstrates substantially higher median WD values compared to MCMC reference values at all sample sizes. Conversely, the shape intercept $\beta_{0\xi}$ exhibits good performance with WD values closely aligned to reference WD metrics across all sample sizes, similarly to shape regression coefficients $\vec{\gamma}_\xi$ in terms of SD.

Variance latent variables (λ_σ^2 and λ_ξ^2) constitute an exception to both the sample size independence pattern

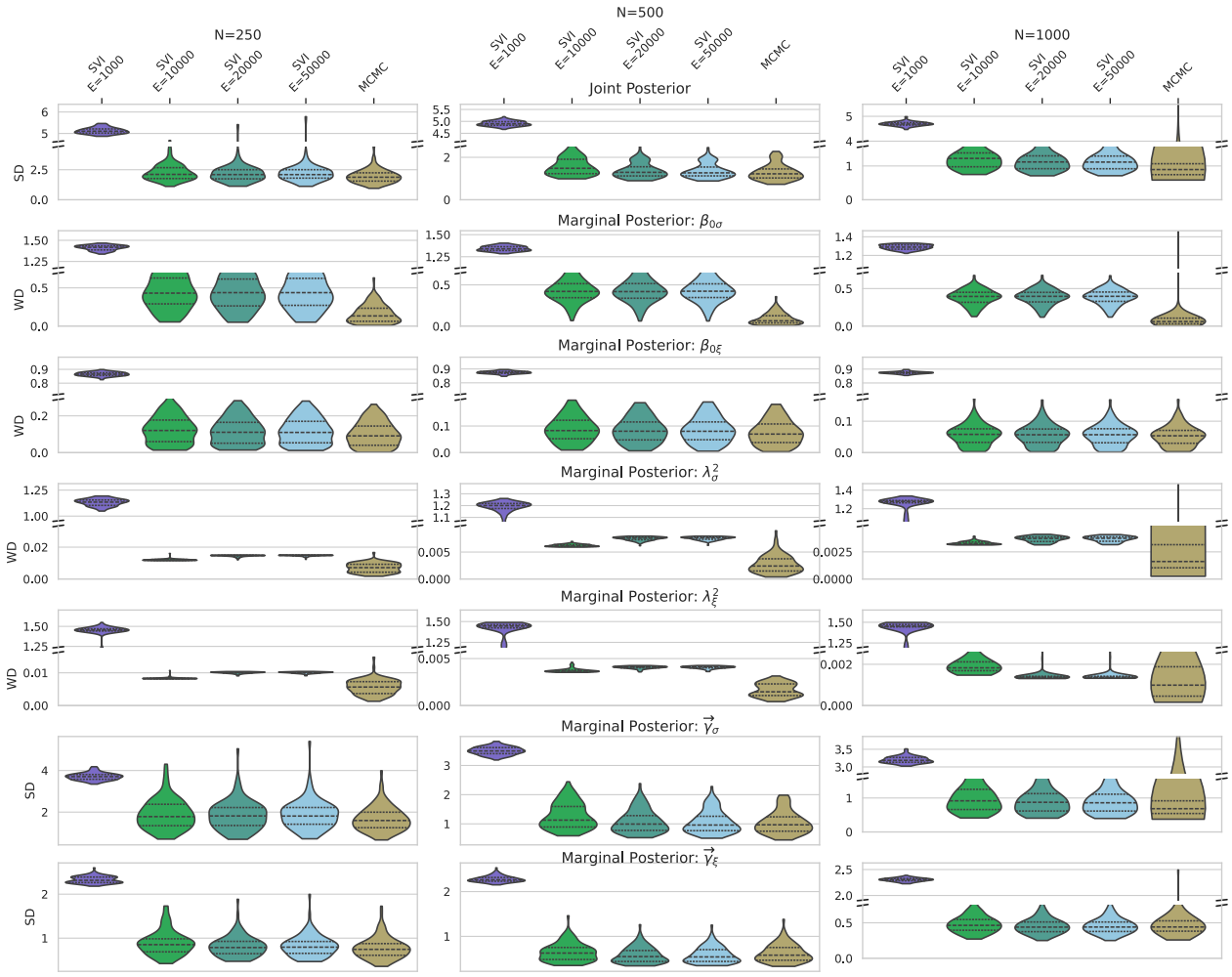
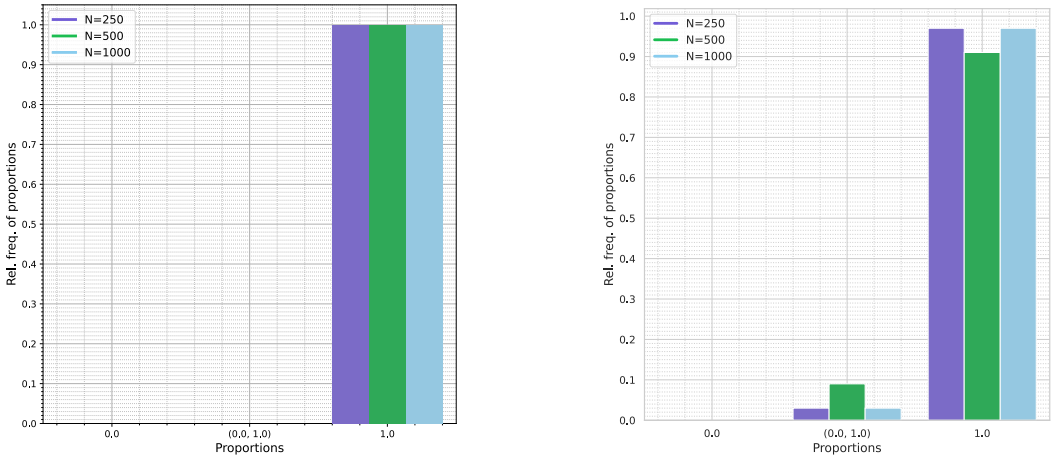


Figure 7.1: SVI Shows Close WD/SD But with a Persistent Differences in Scale and Variance Latent Variables

WD stands for Wasserstein distance. SD stands for Sinkhorn distance which was used to approximate the Wasserstein Distance with an ϵ-regularizer of $\epsilon = 0.01$. The lower the WD or SD, the better. The posteriors contained 32000 samples.

and scale-shape-related pattern. These variables exhibit substantial differences from MCMC reference WD values at $N = 250$ and $N = 500$, but the discrepancy decreases considerably at $N = 1000$, with λ_ξ^2 showing a proportionally greater improvement to λ_σ^2.

OOS behaviour: Support validity tests, given the final inferred optimal parameters, were carried out for 100 SVI with 20000 epochs and MCMC runs of the simulation study. Figure 7.2 indicates that SVI with the modified GPD, in contrast to MCMC with the naïve GPD, never leads to an inferred posterior that results in OOS responses. For MCMC, complete (100%) support validity was achieved in 97%, 91%, and 97% of runs, for $N = 250$, $N = 500$, and $N = 1000$, respectively.



(a) SVI (b) MCMC

Figure 7.2: Incidence of OOS Responses: Lower Rate in SVI with Modified GPD Compared to MCMC with Naïve GPD Across Sample Sizes
"Proportions" refers to number of OOS responses compared to the respective sample size. "Relative frequency of proportions" refers to the relative frequency of the proportions of OOS responses to the respective sample size over the number of SVI and MCMC runs.

The OOS occurrences analysis utilized a single SVI run with 20000 epochs. Two sets of latent variable realizations were evaluated: those sampled S-times from the variational distribution for ELBO gradient estimation, and those sampled once from the optimal variational distribution inferred at each epoch. OOS instances occurred frequently during optimization for ELBO gradient estimation samples, while no OOS events were observed for latent variables sampled from the optimal variational distribution at any epoch (Figure 7.3). The OOS count frequency during ELBO gradient estimation exhibits high initial values that decrease progressively as ELBO convergence occurs.

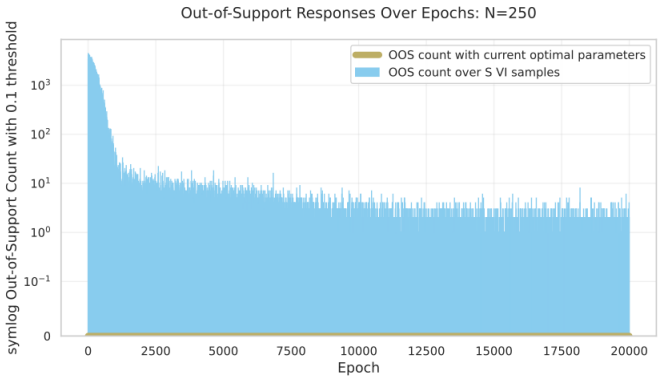
Given a single SVI run for 20000 epochs and for $N = 1000$, the modified GPD produced penalty values for the OOS cases are visible as additional noise to the ELBO (see Figure 7.5), yet the ELBO remains monotone and convergent (see Figure 7.4). It was noticed that the noise severity corresponds directly to the modified GPD's penalty parameter (see Subsection 6.6).

Statistics for WD/SD comparison to MCMC Baseline: 100 runs, N=250										
	SVI E=50000				MCMC					
	0.25	median	0.75	IQR	0.25	median	0.75	IQR	MIQR	|MG|
Joint. P.	1.802	2.095	2.505	0.702	1.555	**1.877**	2.243	0.688	✓	0.218
M. P. $\beta_{0\sigma}$	0.272	0.4378	0.6238	0.3518	0.064	**0.134**	0.2345	0.1706	×	0.3038
M. P. $\beta_{0\xi}$	0.0535	0.1099	0.1703	0.1168	0.0402	**0.0912**	0.1446	0.1045	✓	0.0187
M. P. λ^2_σ	0.01457	0.01485	0.01494	0.00037	0.0043	**0.00716**	0.00925	0.00496	×	0.00769
M. P. λ^2_ξ	0.00992	0.01013	0.01024	0.00032	0.00366	**0.00565**	0.00729	0.00364	×	0.00448
M. P. $\overrightarrow{\gamma}_\sigma$	1.417	1.81	2.221	0.804	1.249	**1.595**	1.996	0.747	✓	0.215
M. P. $\overrightarrow{\gamma}_\xi$	0.654	0.795	0.924	0.27	0.61	**0.745**	0.876	0.265	✓	0.05
Statistics for WD/SD comparison to MCMC Baseline: 100 runs, N=500										
	SVI E=50000				MCMC					
	0.25	median	0.75	IQR	0.25	median	0.75	IQR	MIQR	|MG|
Joint. P.	1.125	1.267	1.542	0.417	1.026	**1.222**	1.452	0.426	✓	0.045
M. P. $\beta_{0\sigma}$	0.3469	0.4234	0.5125	0.1656	0.0401	**0.065**	0.1267	0.0866	×	0.3584
M. P. $\beta_{0\xi}$	0.0485	0.0802	0.1153	0.0669	0.038	**0.0697**	0.1076	0.0696	✓	0.0105
M. P. λ^2_σ	0.00736	0.00763	0.00776	0.0004	0.00149	**0.00242**	0.00375	0.00226	×	0.00521
M. P. λ^2_ξ	0.004	0.00409	0.00416	0.00016	0.00106	**0.00146**	0.0023	0.00124	×	0.00263
M. P. $\overrightarrow{\gamma}_\sigma$	0.767	**0.96**	1.26	0.493	0.749	0.969	1.241	0.492	✓	0.009
M. P. $\overrightarrow{\gamma}_\xi$	0.44	**0.541**	0.704	0.264	0.467	0.58	0.751	0.284	✓	0.039

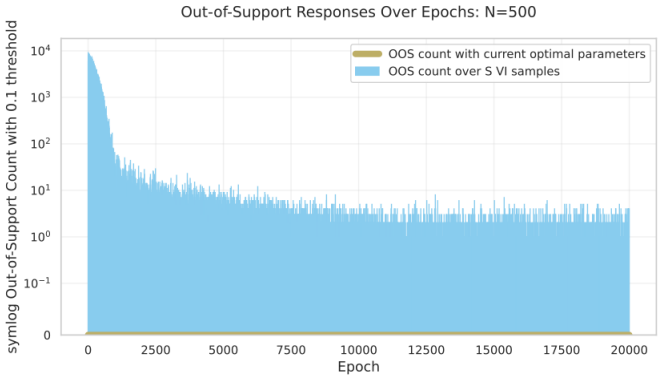
Statistics for WD/SD comparison to MCMC Baseline: 100 runs, N=1000										
	SVI E=50000				MCMC					
	0.25	median	0.75	IQR	0.25	median	0.75	IQR	MIQR	|MG|
Joint. P.	0.917	1.114	1.293	0.375	0.742	**0.893**	1.066	0.323	×	0.221
M. P. $\beta_{0\sigma}$	0.3297	0.3965	0.4536	0.1239	0.0329	**0.0623**	0.1069	0.074	×	0.3342
M. P. $\beta_{0\xi}$	0.0315	0.057	0.076	0.0445	0.0298	**0.0536**	0.0707	0.0409	✓	0.0034
M. P. λ^2_σ	0.00348	0.00375	0.00389	0.00041	0.00104	**0.00161**	0.00316	0.00212	×	0.00214
M. P. λ^2_ξ	0.00135	0.00137	0.00142	0.00007	0.00045	**0.00099**	0.00188	0.00143	×	0.00038
M. P. $\overrightarrow{\gamma}_\sigma$	0.616	0.859	1.112	0.496	0.543	**0.685**	0.917	0.374	✓	0.174
M. P. $\overrightarrow{\gamma}_\xi$	0.378	**0.441**	0.511	0.133	0.383	0.442	0.528	0.145	✓	0.001

Table 7.3: SVI Shows Close WD/SD Quantile Statistics But with a Persistent Differences in Scale and Variance Latent Variables

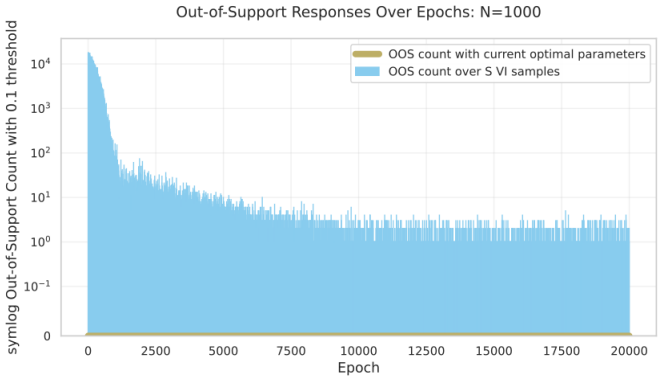
The bold written numbers represent the lowest median value in the respective row. "M. P." stands for Marginal Posterior and "Joint. P." stands for Joint Posterior. IQR represents the interquartile range. MIQR stands for Median in Quartile Range and indicates with a checkmark (✓) that the MCMC median falls within the range of the SVI's 0.25-th and 0.75-th quartile, and with a cross (×) if it does not. |MG| stands for absolute Median Gap and represents the absolute difference between the SVI and MCMC median values.



(a) $N = 250$



(b) $N = 500$



(c) $N = 1000$

Figure 7.3: OOS Cases Frequently Occur When Drawing Samples from the Variational Distribution During Optimization, While the at any Given Epoch Optimal Variational Parameters Show Non-Occurrence of OOS Cases

The y-axis uses a symmetric logarithmic (symlog) scale to display the OOS count. A symlog scale handles both positive, zero, and negative by allowing a linear scale for values below 0.1 and transitioning to logarithmic scaling for values above 0.1. The blue line symbolize the OOS count for the S latent variables that have been sampled from the variational distribution to estimate the ELBO's gradient while the golden line shows the OOS count for the latent variables sampled once from the variational distribution with the optimal parameters inferred at the end of an epoch.

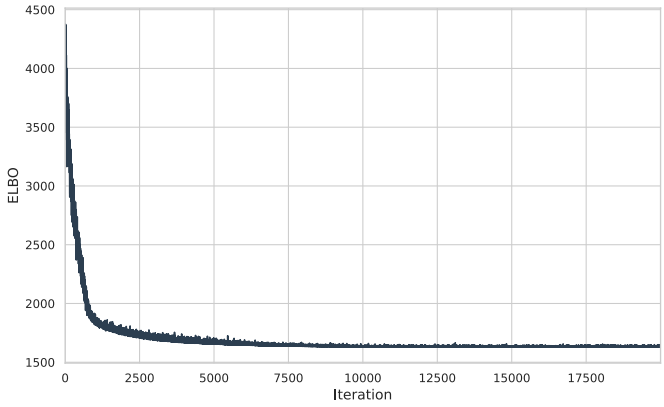


Figure 7.4: ELBO of SVI With a Modified GPD Is Able to Converge: $N = 1000$

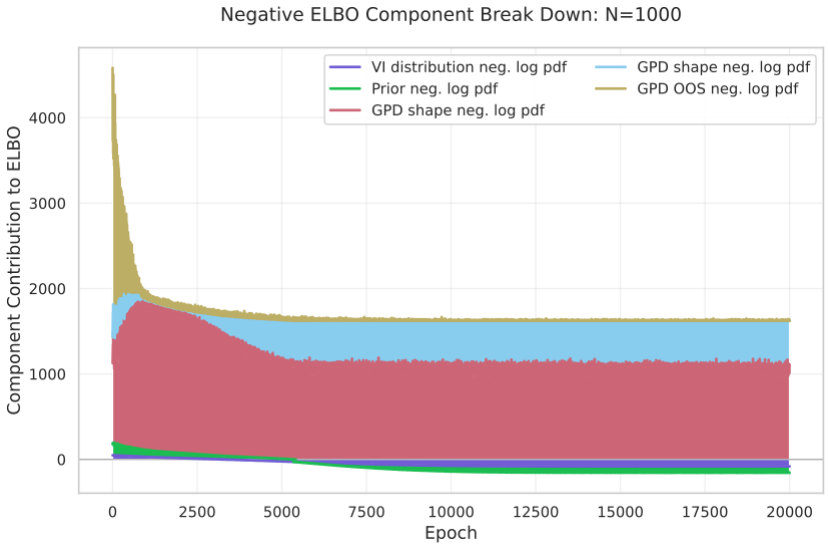


Figure 7.5: Noise in the ELBO During SVI Optimization Results From the Modified GPD Penalizing Detected OOS Cases

Within the legend "neg." stands for negative and "pos." for positive. The components refer to the summation of the log terms within the ELBO over the responses as detailed in Chapter 5. The GPD's log PDF contribution to the ELBO is further disaggregated into responses either within or outside the support. For responses within the support, this is further segmented based on the sign of the estimated shape parameter ξ where "positive" includes 0.

7.2 Case Study

In this case study, the two-stage PoT procedure was applied to the *Danish Fire Insurance* data. A Bayesian SADR AL-based quantile regression first inferred a covariate-dependent threshold at the 0.91 quantile-level. Then a Bayesian SADR GP model was fitted to the exceedances above the threshold with the posterior inferred with SVI using the modified GPD and with MCMC using the naïve GPD. The model's goodness-of-fit was examined utilizing a QQ plot specific to the GPD. The evaluation process included visual evaluation of kernel density estimates for the latent variable marginal posterior distributions of SVI and MCMC, along with visual comparison of their GPD scale and shape parameter estimates, including the HDIs. Execution time measurements for both methods were also recorded.

All hyperparameter settings, initializations, and computational resources are identical to those documented in Chapter 6.

First Stage: At a quantile level of 0.91, Bayesian quantile regression using the ALD model yielded 204 exceedances above the threshold, representing 9.4% of the total 2167 observations. The minimal and maximal threshold values were 4.8 million DKK and 9.7 million DKK, respectively, as illustrated in Figure 7.6.

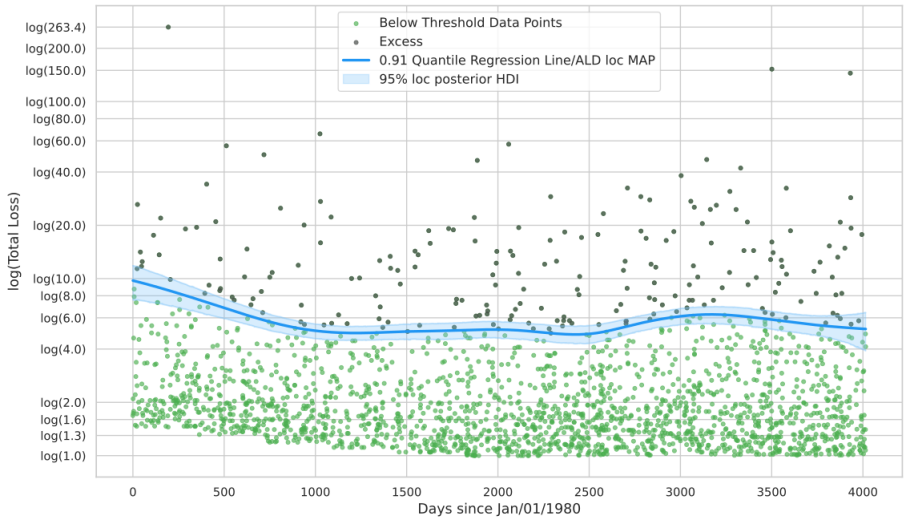


Figure 7.6: ALD Quantile Regression of Total Loss on Days since Jan/01/1980 at the 0.91 Quantile Level

Second Stage: The MCMC and SVI run achieved good convergence diagnostics (see Table 7.4 and Appendix E.1).

Table 7.4: Baseline MCMC Diagnostic Statistics for Various Sample Sizes

Metric	Values
Worst $\hat{R}$	1.001485
ESS BULK (min–max)	Ca. 2753–12845
ESS TAIL (min–max)	Ca. 1781–16064
Funnel of Hell	No

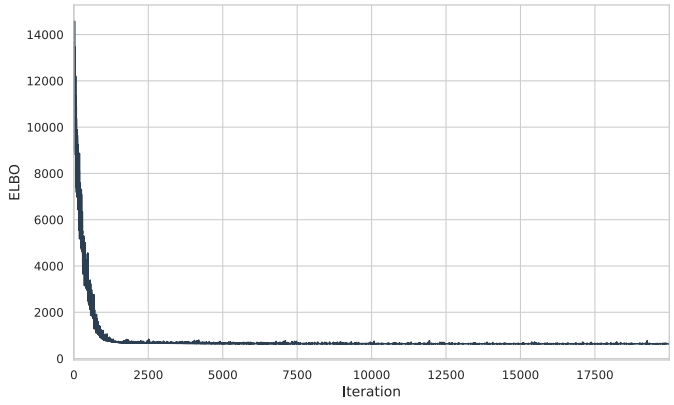


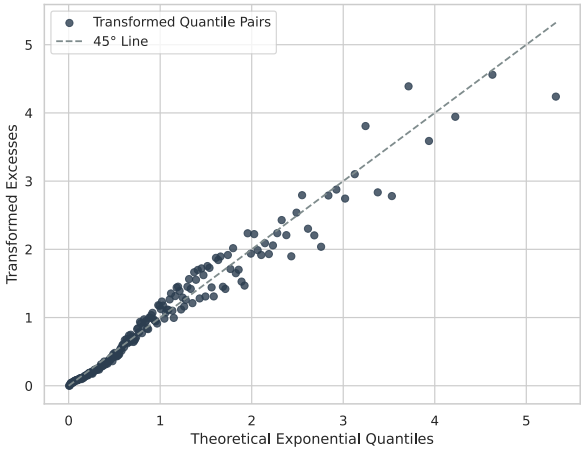
Figure 7.7: The ELBO in the SVI GPD Posterior Inference converged

As shown in Figure 7.8, given a quantile level of $\tau = 0.91$, both SVI and MCMC exhibit identical goodness-of-fit to the theoretical exponential distribution, with points following the 45° line in the central region (0-2) and similar deviations in the upper quantiles. The increased dispersion at higher quantiles shows a greater tail variance without systematic bias since the points remain evenly distributed around the reference line.

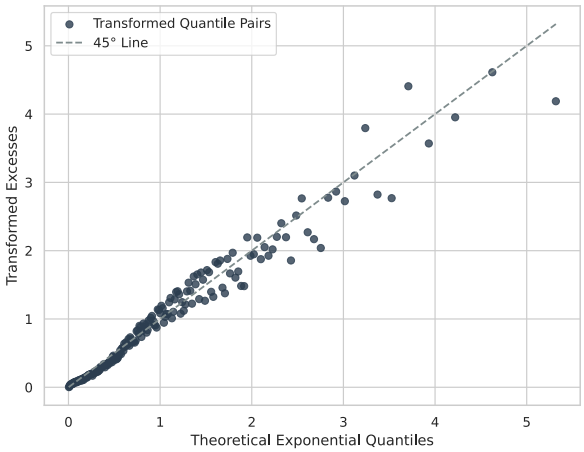
The kernel density estimate plots (Figure 7.9) illustrate the different degrees of alignment between the marginal posteriors of the structured additive predictors' latent variables inferred by SVI and MCMC.

Comparison between SVI and MCMC inferred latent variable estimates reveals distinctive patterns across latent variable categories. Shape SAP regression coefficient latent variables (γ_{1-21}) exhibit strong modal alignment across all coefficients, but with SVI consistently underestimating posterior variance. The shape intercept latent variable (β_0) demonstrates robust modal alignment between SVI and MCMC methodologies with minimal underestimation of variance by SVI. For the shape variance latent variable (λ^2), SVI produces a slightly offset mode that remains within the MCMC distribution's high-density region, though SVI substantially underestimates posterior variance.

Scale SAP regression coefficient latent variables (γ_{1-21}) display modal locations with minor deviations from MCMC estimates, though these differences remain modest. SVI moderately underestimates the posterior variance for most scale coefficients, while select coefficients exhibit satisfactory variance align-



(a) SVI



(b) MCMC

Figure 7.8: SVI and MCMC Demonstrate Equivalent Goodness-of-Fit

ment. The scale intercept (β_0) demonstrates precise correspondence in both modal location and posterior spread between SVI and MCMC methods. For the scale variance latent variable (λ^2), SVI substantially underestimates posterior variance compared to MCMC, yielding a narrow posterior distribution despite relatively aligned modal locations.

Overall, SVI performs adequately in estimating modal locations across most latent variables, with shape latent variables showing better alignment than scale latent variables. The most visible difference is SVI's tendency to underestimate the posterior uncertainty, which is more pronounced for the latent variance variable (λ^2).

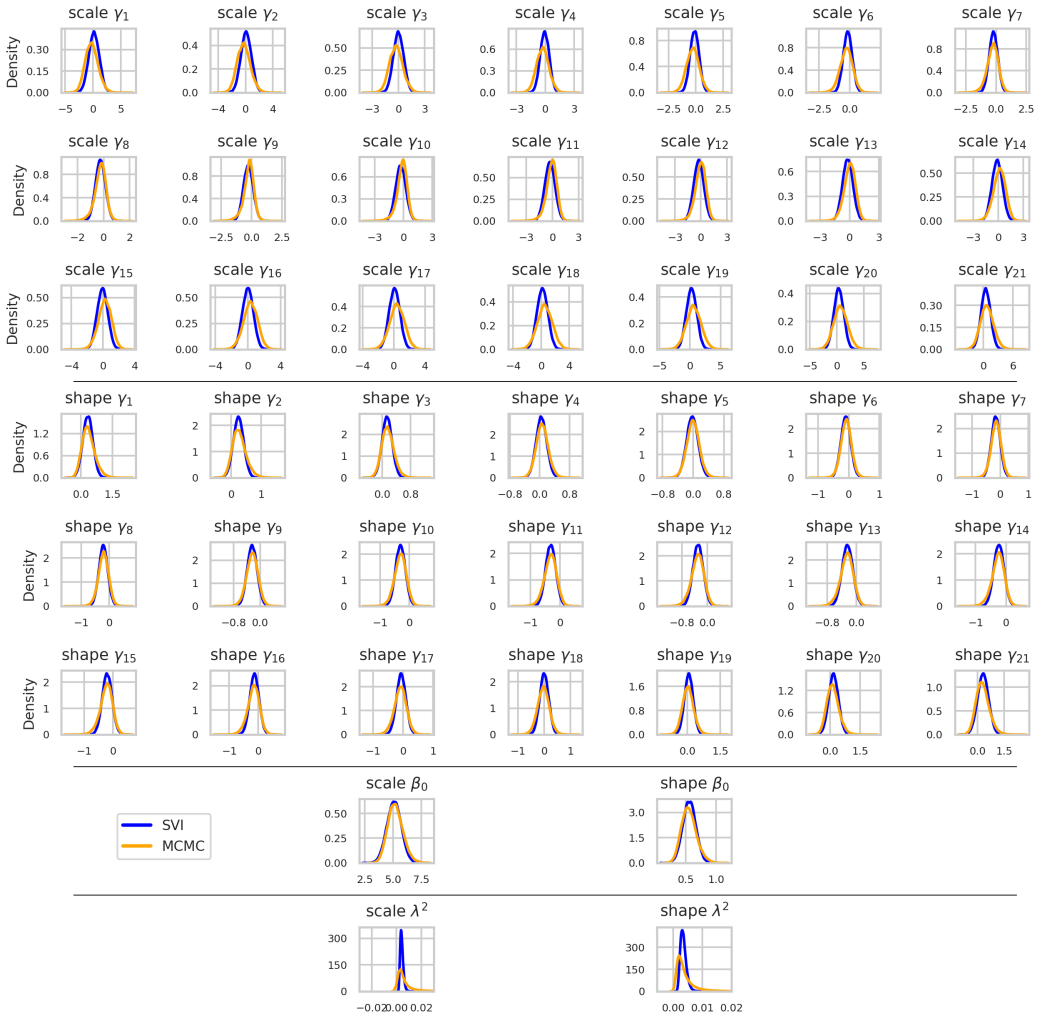


Figure 7.9: SVI Underestimates Posterior Uncertainty Across All Latent Variable Types with the Underestimation Being Most Severe for Variance Latent Variables

Given Figure 7.10, compared to MCMC, SVI infers shape point estimates that closely follow the MCMC

estimates, but the scale point estimates deviate slightly but notably. Regarding the estimated uncertainty, SVI underestimates the uncertainty, particularly at the limits of the observation period.

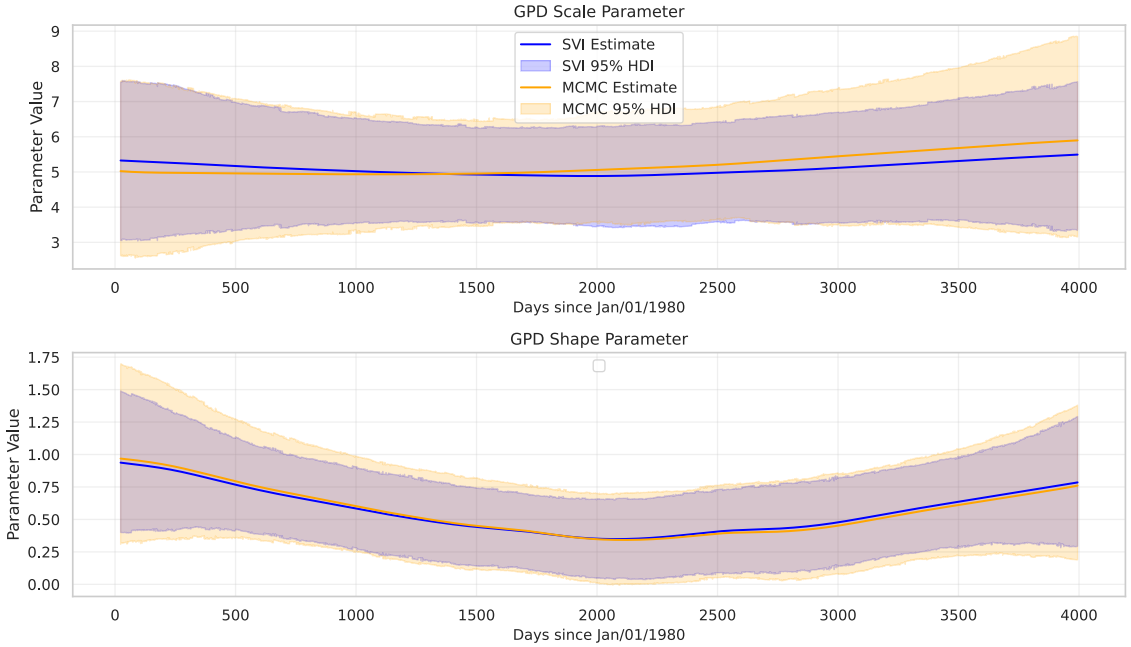


Figure 7.10: SVI Achieves Consistency with MCMC for GPD Shape Parameter Point Estimates, but Deviates for the Scale and Underestimates Uncertainty

SVI computation with 20000 epochs required 16.0 seconds, while MCMC processing with 6000 warm-up iterations and 12000 posterior samples required 42 : 12 minutes.

Chapter 8

Discussion

This thesis investigated the applicability of SVI equipped with a FCMN variational distribution and a modified univariate GPD response distribution for posterior estimation in SADR models compared to MCMC with a naïve univariate GPD. As explained in Chapter 1, a standard/naïve GPD introduces computational challenges for SVI when observations fall outside the parameter-dependent support, leading to undefined gradients that impede optimization. The modified GPD implementation described in Section 6.6 addresses this issue by returning finite values for OOS responses, which implicitly penalize parameter values that cause OOS cases.

The comparative analysis was conducted through both simulation and empirical case studies with a one-dimensional covariate. The simulation study, detailed in Section 6.8, systematically evaluated SVI and MCMC on simulated GP-distributed data with covariate-dependent scale and shape parameters across varying sample sizes ($N = 250, N = 500, N = 1000$). The WD and its SD approximation quantified the differences between posterior distributions as described in Section 6.8.5. The case study applied a two-stage PoT procedure to the *Danish Fire Insurance* dataset, implementing Bayesian quantile regression with ALD to determine thresholds followed by fitting the GPD for exceedances, as outlined in Section 6.9.

Regarding OOS behavior, the simulation study showed that SVI with the modified GPD never led to an inferred posterior that results in OOS responses, while MCMC with the naïve GPD occasionally produced parameter estimates leading to OOS responses. MCMC achieved complete support validity in 97%, 91%, and 97% of runs for $N = 250$, $N = 500$, and $N = 1000$, respectively. Although OOS instances occurred frequently when drawing variational samples during ELBO gradient estimation in SVI optimization, the modified GPD effectively handled these cases without compromising convergence, as evidenced by the consistent monotonic trajectory of the ELBO despite the introduced penalty noise.

Further, the simulation study results (see Section 7.1) indicate that SVI demonstrates adequate approximation quality to MCMC in terms of SD for the joint distribution. At sample sizes $N = 250$ and $N = 500$, SVI exhibits SD values proximate to MCMC reference metrics, while performance marginally degrades at $N = 1000$.

GP scale related SAP latent variables persistently deviated from the MCMC reference. Especially the intercept $\beta_{0\sigma}$ latent variable demonstrated substantial discrepancies across all sample sizes. The scale regression coefficients $\overrightarrow{\gamma}_\sigma$ showed modest SD discrepancies that remained across sample sizes too. Conversely, the shape intercept $\beta_{0\xi}$ exhibited good performance with WD values closely aligned to reference WD metrics across all sample sizes, similarly to shape regression coefficients $\overrightarrow{\gamma}_\xi$ in terms of SD.

The variance latent variables (λ^2_σ and λ^2_ξ) exhibited substantial differences from MCMC reference WD values at $N = 250$ and $N = 500$, but the discrepancy decreased considerably at $N = 1000$, with λ^2_ξ showing a proportionally greater improvement compared to λ^2_σ.

For SADR with GP response distribution, SVI demonstrates adequate approximation capabilities for both joint and shape posteriors, while exhibiting suboptimal performance regarding scale SAP latent variables. These scale parameter approximation limitations cannot be attributed to sample size constraints. The variance latent variable display suboptimal approximation capabilities at low sample sizes, though the approximation quality improves with increasing sample size. WD/SD discrepancies observed for scale SAP parameters in the simulation study manifest in the case study as systematic biases and uncertainty underestimation.

The *Danish Fire Insurance* case study results revealed comparable model fit between methods. The QQ plots in Figure 7.8 demonstrated equivalent goodness-of-fit to the theoretical exponential distribution. However, kernel density estimates in Figure 7.9 showed that SVI consistently underestimated posterior uncertainty across all latent variable types, with the impact most pronounced for the SAPs variance latent variables. This uncertainty underestimation represents a limitation of the variational approach despite its computational advantages. For point estimates of GPD parameters shown in Figure 7.10, SVI achieved consistency with MCMC for the shape parameter, but showed notable biases for the scale parameter.

The observed biases are not necessarily a systemic inferiority of SVI. Although no relative accuracy statement of SVI to MCMC has been derived (Blei et al., 2017), it has been shown that SVI can be accurate in inferring posterior densities (Blei & Jordan, 2006; Braun & McAuliffe, 2010; Kucukelbir et al., 2016). Yet, the inherent underestimation of variance is a systemic issue in SVI that uses the exclusive KL divergence (Blei et al., 2017; Minka, 2005), which is used in this thesis. Possible alternatives include employing divergence terms other than the exclusive KL-divergence, such as Rényi's α-divergence and χ-divergence by Li and Turner (2016) and Dieng et al. (2017), respectively. However, in a moderate to high-dimensional problem setting as in this thesis, Dhaka et al. (2021) recommend maintaining the exclusive KL-divergence for optimization stability and increasing accuracy by rather opting for a more expressive variational distribution or reparameterizing the model. With respect to the usage of a more expressive variational distribution, Agrawal et al. (2020) conclude that the FCMN variational distribution's performance is enhanced when combined with importance-weighted sampling. Still, it falls short compared to the more flexible normalizing flows (real NVP), which provide substantial accuracy improvements when combined with appropriate gradient estimators. With regard to the suggestion of Dhaka et al. to reparameterize the model, the use of extremely high `target_accept` rates of 0.999

and 0.99, and a `max_treedepth` of 20 for the simulation and case study, respectively, indicates that the SADR model with a GP response has a challenging posterior geometry with regions of high curvature (Betancourt, 2018). For such posteriors, Betancourt (2018) recommends using more informative priors, or alternatively using a different parameterization. However, more informative priors than those adopted in this thesis are not suitable. Evaluation of alternative prior specifications for variance latent variables is recommended. The SADR framework otherwise presents restricted reparameterization possibilities.

Regarding computational efficiency, SVI completed in 16.0 seconds compared to MCMC's $42:12$ minutes—a substantial performance difference that was to be expected.

Chapter 9

Conclusion

This thesis examined the application of SVI with a modified GP distribution for Bayesian SADR models compared to MCMC with a naïve GP distribution. The investigation addressed the computational challenge posed by the GP distribution's parameter-dependent support, which typically results in undefined gradients during SVI optimization when observations fall outside the support.

The simulation study demonstrated that SVI with the modified GP distribution effectively prevents OOS cases in the final inferred posterior, while MCMC with the naïve GP occasionally produces parameter estimates leading to OOS responses ($3-9\%$ of runs, depending on the sample size). Although OOS instances occurred during SVI optimization when drawing samples for gradient estimation, the modified GP successfully handled these cases without compromising convergence.

Performance evaluation through WD and SD metrics revealed that SVI provides an adequate approximation to the joint posterior distribution, with quality comparable to MCMC in smaller ($N = 250, N = 500$) and a slight decrease in larger sample sizes ($N = 1000$). Examination of individual latent variable groups showed that SVI approximates shape-related SAP intercept and regression coefficients well across all sample sizes but exhibits persistent discrepancies for scale-related SAP latent variables. Variance parameters displayed substantial differences from MCMC at smaller sample sizes. The approximation quality improved considerably with an increased sample size.

The *Danish Fire Insurance* case study confirmed these patterns. Both methods achieved equivalent goodness-of-fit to the theoretical exponential distribution as shown by the QQ plots. However, SVI consistently underestimated posterior uncertainty across all latent variable types. For point estimates, SVI showed good consistency with MCMC for shape parameters, but notable biases for scale parameters.

The observed variance underestimation represents a known limitation of variational methods using the exclusive KL divergence. While alternative divergence measures or more expressive variational distributions could potentially address this limitation, they introduce additional computational complexity. The computational efficiency advantage of SVI (16.0 seconds versus $42:12$ minutes for MCMC) represents a

substantial practical benefit for applications where rapid inference is prioritized over precise uncertainty quantification.

Future research directions include exploring more expressive variational distributions, such as normalizing flows, and investigating alternative prior specifications for the SADR variance latent variables.

Bibliography

Agrawal, A., Sheldon, D., & Domke, J. (2020). Advances in black-box vi: Normalizing flows, importance weighting, and optimization. https://arxiv.org/abs/2006.10343

Ashkar, F., & Nwentsa Tatsambon, C. (2007). Revisiting some estimation methods for the generalized Pareto distribution. *Journal of Hydrology*, *346*(3-4), 136–143. https://doi.org/10.1016/j.jhydrol.2007.09.007

Attias, H. (1999). Inferring parameters and structure of latent variable models by variational bayes. *UAI'99: Proceedings of the Fifteenth Conference on Uncertainty in Artificial Intelligence*, 21–30. https://dl.acm.org/doi/10.1145/168304.168306

Balkema, A. A., & de Haan, L. (1974). Residual life time at great age. *The Annals of Probability*, *2*(5), 792–804. https://doi.org/10.1214/aop/1176996548

Barber, D., & Wiegerinck, W. (1998). Tractable variational structures for approximating graphical models. In M. Kearns, S. Solla, & D. Cohn (Eds.), *Advances in neural information processing systems* (Vol. 11). MIT Press. https://proceedings.neurips.cc/paper_files/paper/1998/file/297fa7777981f402dbba17e9f29e292d-Paper.pdf

Betancourt, M. (2018). A conceptual introduction to hamiltonian monte carlo. https://arxiv.org/abs/1701.02434

Bishop, C. M. (2006). *Pattern recognition and machine learning*. Springer.

Blei, D. M., & Jordan, M. I. (2006). Variational inference for Dirichlet process mixtures. *Bayesian Analysis*, *1*(1), 121–143. https://doi.org/10.1214/06-BA104

Blei, D. M., Kucukelbir, A., & McAuliffe, J. D. (2017). Variational Inference: A Review for Statisticians. *Journal of the American Statistical Association*, *112*(518), 859–877. https://doi.org/10.1080/01621459.2017.1285773

Bourguignon, J.-P., Jeltsch, R., Pinto, A., & Viana, M. (2015, January). *Mathematics of energy and climate change: International conference and advanced school planet earth, portugal, march 21-28, 2013* (Vol. 2). https://doi.org/10.1007/978-3-319-16121-1

Bradbury, J., Frostig, R., Hawkins, P., Johnson, M. J., Leary, C., Maclaurin, D., Necula, G., Paszke, A., VanderPlas, J., Wanderman-Milne, S., & Zhang, Q. (2018). JAX: Composable transformations of Python+NumPy programs. *Google*. http://github.com/google/jax

Braun, M., & McAuliffe, J. (2010). Variational inference for large-scale models of discrete choice. *Journal of the American Statistical Association*, *105*(489), 324–335. https://doi.org/10.1198/jasa.2009.tm08030

Callegher, G., Kneib, T., Söding, J., & Wiemann, P. (2025). Stochastic Variational Inference for Structured Additive Distributional Regression. *arXiv*. https://arxiv.org/abs/2412.10038

Carlin, B. P., & Louis, T. A. (2009). *Bayesian methods for data analysis* (Third edition). CRC Press.

Castillo, E., & Hadi, A. S. (1997). Fitting the generalized Pareto distribution to data. *Journal of the American Statistical Association*, *92*(440), 1609–1620. https://doi.org/10.1080/01621459.1997.10473683

Coles, S. (2001). *An Introduction to Statistical Modeling of Extreme Values*. Springer London. https://doi.org/10.1007/978-1-4471-3675-0

Courant, R., & Hilbert, D. (1989). *Methods of mathematical physics* (First Edition). Wiley-VCH Verlag GmbH & Co. KGaA. https://doi.org/10.1002/9783527617210

Cuturi, M. (2013). Sinkhorn distances: Lightspeed computation of optimal transport [Editors: C.J. Burges and L. Bottou and M. Welling and Z. Ghahramani and K.Q. Weinberger]. *Advances in Neural Information Processing Systems*, *26*. https://proceedings.neurips.cc/paper_files/paper/2013/file/af21d0c97db2e27e13572cbf59eb343d-Paper.pdf

Cuturi, M., Meng-Papaxanthos, L., Tian, Y., Bunne, C., Davis, G., & Teboul, O. (2022). Optimal transport tools (ott): A jax toolbox for all things wasserstein. *arXiv*. http://arxiv.org/abs/2201.12324v1

Das, S., Harel, O., Dey, D. K., Covault, J., & Kranzler, H. R. (2010). Analysis of extreme drinking in patients with alcohol dependence using Pareto regression. *Statistics in Medicine*. https://doi.org/10.1002/sim.3878

de Boor, C. (2001). *A practical guide to splines* (First Edition). Springer. https://www.amazon.com/Practical-Guide-Splines-Mathematical-Sciences/dp/0387953663

de Zea Bermudez, P., & Kotz, S. (2010a). Parameter estimation of the generalized Pareto distribution—Part I. *Journal of Statistical Planning and Inference*, *140*, 1353–1373. https://doi.org/10.1016/j.jspi.2008.11.019

de Zea Bermudez, P., & Kotz, S. (2010b). Parameter estimation of the generalized Pareto distribution—Part II. *Journal of Statistical Planning and Inference*, *140*, 1374–1388. https://doi.org/10.1016/j.jspi.2008.11.020

DeepMind, Babuschkin, I., Baumli, K., Bell, A., Bhupatiraju, S., Bruce, J., Buchlovsky, P., Budden, D., Cai, T., Clark, A., Danihelka, I., Dedieu, A., Fantacci, C., Godwin, J., Jones, C., Hemsley, R., Hennigan, T., Hessel, M., Hou, S., … Viola, F. (2020). *The DeepMind JAX Ecosystem*. http://github.com/google-deepmind

Dhaka, A. K., Catalina, A., Welandawe, M., Andersen, M. R., Huggins, J. H., & Vehtari, A. (2021). Challenges and opportunities in high-dimensional variational inference. *arXiv*. http://arxiv.org/abs/2103.01085v1

Dieng, A. B., Tran, D., Ranganath, R., Paisley, J., & Blei, D. M. (2017). Variational inference via χ upper bound minimization. *Proceedings of the 31st International Conference on Neural Information Processing Systems*, 2729–2738.

Dillon, J. V., Langmore, I., Tran, D., Brevdo, E., Vasudevan, S., Moore, D., Patton, B., Alemi, A., Hoffman, M., & Saurous, R. A. (2017). Tensorflow distributions. *arXiv*. http://arxiv.org/abs/1711.10604v1

Domke, J., Garrigos, G., & Gower, R. (2023). Provable convergence guarantees for black-box variational inference. *arXiv*. http://arxiv.org/abs/2306.03638

Dupuis, D. J., & Tsao, M. (1998). A hybrid estimator for generalized pareto and extreme-value distributions. *Communications in Statistics - Theory and Methods*, *27*(4), 925–941. https://doi.org/10.1080/03610929808832136

Eilers, P. H. C., & Marx, B. D. (1996). Flexible smoothing with B-splines and penalties. *Statistical Science*, *11*(2), 89–121. https://doi.org/10.1214/ss/1038425655

Embrechts, P., Klüppelberg, C., & Mikosch, T. (1997). *Modelling extremal events: For insurance and finance* (1st ed., Vol. 33). Springer Berlin Heidelberg. https://doi.org/10.1007/978-3-642-33483-2

Fahrmeir, L., Kneib, T., Lang, S., & Marx, B. D. (2013). *Regression: Models, methods and applications* (1st). Springer Berlin Heidelberg. https://doi.org/10.1007/978-3-642-34333-9

Fisher, R. A., & Tippett, L. H. C. (1928). Limiting forms of the frequency distribution of the largest or smallest member of a sample. *Mathematical Proceedings of the Cambridge Philosophical Society*, *24*(2), 180–190. https://doi.org/10.1017/S0305004100015681

Flamary, R., Courty, N., Gramfort, A., Alaya, M. Z., Boisbunon, A., Chambon, S., Chapel, L., Corenflos, A., Fatras, K., Fournier, N., Gautheron, L., Gayraud, N. T., Janati, H., Rakotomamonjy, A., Redko, I., Rolet, A., Schutz, A., Seguy, V., Sutherland, D. J., … Vayer, T. (2021). Pot: Python optimal transport. *Journal of Machine Learning Research*, *22*(78), 1–8. http://jmlr.org/papers/v22/20-451.html

Gelman, A., Carlin, J. B., Stern, H. S., Dunson, D. B., Vehtari, A., & Rubin, D. B. (2014). *Bayesian data analysis* (Third edition). CRC Press.
OCLC: 909477393.

Geman, S., & Geman, D. (1984). Stochastic relaxation, gibbs distributions, and the bayesian restoration of images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *PAMI-6*(6), 721–741. https://doi.org/10.1109/TPAMI.1984.4767596

Ghahramani, Z., & Beal, M. J. (2000). Propagation algorithms for variational bayesian learning. *Proceedings of the 14th International Conference on Neural Information Processing Systems*, 486–492.

Gnedenko, B. V. (1943). Sur la distribution limite du terme maximum d'une série aléatoire [the limiting distribution of the maximum term of a random sequence]. *Annals of Mathematics*, *44*(3), 423–453.

Hastie, T., & Tibshirani, R. (1986). Generalized additive models. *Statistical Science*, *1*(3), 297–310. https://doi.org/10.1214/ss/1177013604

Hinton, G. E., & van Camp, D. (1993). Keeping the neural networks simple by minimizing the description length of the weights. *COLT '93: Proceedings of the Sixth Annual Conference on Computational Learning Theory*, 5–13. https://doi.org/10.1145/168304.168306

Hoffman, M. D., Blei, D. M., Wang, C., & Paisley, J. (2013). Stochastic variational inference. *Journal of Machine Learning Research*, *14*(40), 1303–1347. http://jmlr.org/papers/v14/hoffman13a.html

Holbrook, A. (2018). Differentiating the pseudo determinant. *Linear Algebra and its Applications*, *548*, 293–304. https://doi.org/10.1016/j.laa.2018.03.018

Homan, M. D., & Gelman, A. (2014). The no-u-turn sampler: Adaptively setting path lengths in hamiltonian monte carlo. *J. Mach. Learn. Res.*, *15*(1), 1593–1623.

Jaakkola, T. S., & Jordan, M. I. (2000). Bayesian parameter estimation via variational methods. *Statistics and Computing*, *10*, 25–37. https://doi.org/10.1023/A:1008932416310

Jordan, M. I., Ghahramani, Z., Jaakkola, T. S., & Saul, L. K. (1999). An introduction to variational methods for graphical models. *Machine Learning*, *37*, 183–233. https://doi.org/10.1023/A:1007665907178

Kingma, D. P., & Ba, J. L. (2017). Adam: A method for stochastic optimization. *arXiv*. http://arxiv.org/abs/1412.6980v9

Kingma, D. P., & Welling, M. (2013). Auto-Encoding Variational Bayes. *arXiv*. http://arxiv.org/abs/1312.6114

Klein, N., & Kneib, T. (2016). Scale-dependent priors for variance parameters in structured additive distributional regression. *Bayesian Analysis*, *11*(4), 1071–1106. https://doi.org/10.1214/15-BA983

Kleinemeier, J., & Klein, N. (2023). Scalable estimation for structured additive distributional regression through variational inference. *arXiv:2311.07371 [stat.ME]*. https://doi.org/10.48550/arXiv.2311.07371

Kneib, T., Silbersdorff, A., & Säfken, B. (2023). Rage Against the Mean – A Review of Distributional Regression Approaches. *Econometrics and Statistics*, *26*, 99–123. https://doi.org/10.1016/j.ecosta.2021.07.006

Konzen, E., Neves, C., & Jonathan, P. (2021). Modeling nonstationary extremes of storm severity: Comparing parametric and semiparametric inference. *Environmetrics*, *32*, e2667. https://doi.org/10.1002/env.2667

Kucukelbir, A., Tran, D., Ranganath, R., Gelman, A., & Blei, D. M. (2016). Automatic differentiation variational inference. https://arxiv.org/abs/1603.00788

Kullback, S., & Leibler, R. A. (1951). On information and sufficiency. *Annals of Mathematical Statistics*, *22*(1), 79–86. https://doi.org/10.1214/aoms/1177729694

Kumar, R., Carroll, C., Hartikainen, A., & Martin, O. (2019). Arviz a unified library for exploratory analysis of bayesian models in python. *Journal of Open Source Software*, *4*(33), 1143. https://doi.org/10.21105/joss.01143

Lang, S., & Brezger, A. (2004). Bayesian p-splines. *Journal of Computational and Graphical Statistics*, *13*(1), 183–212. https://doi.org/10.1198/1061860043010

Li, Y., & Turner, R. E. (2016). Rényi divergence variational inference. *Proceedings of the 30th International Conference on Neural Information Processing Systems*, 1081–1089.

Liu, Q., & Wang, D. (2016). Stein Variational Gradient Descent: A General Purpose Bayesian Inference Algorithm. *arXiv*. https://arxiv.org/abs/1608.04471v3

MacKay, D. J. C. (1995). Developments in probabilistic modelling with neural networks — ensemble learning. In *Neural networks: Artificial intelligence and industrial applications* (pp. 191–198). Springer. https://doi.org/10.1007/978-1-4471-3087-1_37

Minka, T. (2005, December). *Divergence measures and message passing* (tech. rep. No. MSR-TR-2005-173). Microsoft Research Ltd. Cambridge, UK.

Murphy, K. P. (2023). *Probabilistic machine learning: Advanced topics*. The MIT Press.

Nakajima, S., Watanabe, K., & Sugiyama, M. (2019). *Variational bayesian learning theory*. Cambridge University Press. https://doi.org/10.1017/9781139879354

Oh, H.-S., Lee, T. C. M., & Nychka, D. W. (2011). Fast nonparametric quantile regression with arbitrary smoothing methods. *Journal of Computational and Graphical Statistics*, *20*(2), 510–526. https://doi.org/10.1198/jcgs.2010.10063

Opper, M., & Winther, O. (1996). A mean field algorithm for bayes learning in large feed-forward neural networks. In M. Mozer, M. Jordan, & T. Petsche (Eds.), *Advances in neural information processing systems* (Vol. 9). MIT Press. https://proceedings.neurips.cc/paper_files/paper/1996/file/193002e668758ea9762904da1a22337c-Paper.pdf

Paisley, J., Blei, D. M., & Jordan, M. I. (2012). Variational bayesian inference with stochastic search. *Proceedings of the 29th International Coference on International Conference on Machine Learning*, 1363–1370.

Pascanu, R., Mikolov, T., & Bengio, Y. (2012). On the difficulty of training recurrent neural networks. *arXiv*. http://arxiv.org/abs/1211.5063v1

Peyré, G., & Cuturi, M. (2020). Computational optimal transport. *arXiv*. http://arxiv.org/abs/1803.00567v4

Pfaff, B., Zivot, E., McNeil, A., & Stephenson, A. (2018). *Evir: Extreme values in r* [R package version 1.7-4]. https://doi.org/10.32614/CRAN.package.evir

Pickands III, J. (1975). Statistical inference using extreme order statistics. *The Annals of Statistics*, *3*(1), 119–131. https://doi.org/10.1214/aos/1176343002

Pinheiro, J., & Bates, D. (1996). Unconstrained parametrizations for variance-covariance matrices. *Statistical Computing*, *6*, 289–296. https://doi.org/10.1007/BF00140873

Randell, D., Turnbull, K., Ewans, K., & Jonathan, P. (2016). Bayesian inference for nonstationary marginal extremes. *Environmetrics*, *27*(7), 439–450. https://doi.org/https://doi.org/10.1002/env.2403

Ranganath, R., Gerrish, S., & Blei, D. (2014). Black Box Variational Inference (S. Kaski & J. Corander, Eds.). *Proceedings of Machine Learning Research*, *33*, 814–822. https://proceedings.mlr.press/v33/ranganath14.html

Riebl, H., Wiemann, P. F. V., & Kneib, T. (2023). Liesel: A probabilistic programming framework for developing semi-parametric regression models and custom bayesian inference algorithms. https://arxiv.org/abs/2209.10975

Rigby, R. A., & Stasinopoulos, D. M. (2005). Generalized additive models for location, scale and shape [First published: 14 April 2005]. *Journal of the Royal Statistical Society: Series C (Applied Statistics)*, *54*(3), 507–554. https://doi.org/10.1111/j.1467-9876.2005.00510.x

Robbins, H., & Monro, S. (1951). A stochastic approximation method. *Annals of Mathematical Statistics*, *22*(3), 400–407. https://doi.org/10.1214/aoms/1177729586

Roeder, G., Wu, Y., & Duvenaud, D. (2017). Sticking the Landing: Simple, Lower-Variance Gradient Estimators for Variational Inference. *arXiv*. http://arxiv.org/abs/1703.09194

Saul, L., & Jordan, M. (1995). Exploiting tractable substructures in intractable networks. In D. Touretzky, M. Mozer, & M. Hasselmo (Eds.), *Advances in neural information processing systems* (Vol. 8). MIT Press. https://proceedings.neurips.cc/paper_files/paper/1995/file/285f89b802bcb2651801455c86d78f2a-Paper.pdf

Singmann, H. (2016, March). Hierarchical mpt in stan: Dealing with convergent transitions via control arguments [Accessed: 2025-04-25]. http://singmann.org/hierarchical-mpt-in-stan-i-dealing-with-convergent-transitions-via-control-arguments/

Sjölund, J. (2023). A Tutorial on Parametric Variational Inference. *arXiv*. http://arxiv.org/abs/2301.01236

Stan Development Team. (2024). *Stan reference manual* [Version 2.36, section "Thinning samples"]. https://mc-stan.org/docs/reference-manual/analysis.html%5C#thinning-samples

Stasinopoulos, M. D., Kneib, T., Klein, N., Mayr, A., & Heller, G. Z. (2024). *Generalized additive models for location, scale, and shape: A distributional regression approach, with applications*. Cambridge University Press. https://doi.org/10.1017/9781009410076

Stringer, A. (2023). Identifiability constraints in generalized additive models. *Canadian Journal of Statistics*, *52*(2), 461–476. https://doi.org/10.1002/cjs.11786

Wainwright, M. J., & Jordan, M. I. (2007). Graphical Models, Exponential Families, and Variational Inference. *Foundations and Trends® in Machine Learning*, *1*(1–2), 1–305. https://doi.org/10.1561/2200000001

Waskom, M. L. (2021). Seaborn: Statistical data visualization. *Journal of Open Source Software*, *6*(60), 3021. https://doi.org/10.21105/joss.03021

Wiecki, T. (2017, February). Why hierarchical models are awesome, tricky, and bayesian [Accessed: 2025-04-02]. https://twiecki.io/blog/2017/02/08/bayesian-hierchical-non-centered/

Wiemann, P. F. . (2024). Using the softplus function to construct alternative link functions in generalized linear models and beyond. *Statistical Papers*, *65*, 3155–3180. https://doi.org/10.1007/s00362-023-01509-x

Wood, S. N. (2017). *Generalized additive models: An introduction with r* (2nd ed.) [Second edition; framework based on penalized regression splines]. Chapman; Hall/CRC. https://doi.org/10.1201/9781315370279

Xu, M., Quiroz, M., Kohn, R., & Sisson, S. A. (2018). Variance reduction properties of the reparameterization trick. *arXiv*. http://arxiv.org/abs/1809.10330

Youngman, B. D. (2019). Generalized Additive Models for Exceedances of High Thresholds With an Application to Return Level Estimation for U.S. Wind Gusts. *Journal of the American Statistical Association*, *114*(528), 1865–1879. https://doi.org/10.1080/01621459.2018.1529596

Youngman, B. D. (2020). Evgam: An R package for Generalized Additive Extreme Value Models. *arXiv*. https://arxiv.org/abs/2003.04067

Yu, K., & Moyeed, R. A. (2001). Bayesian quantile regression. *Statistics & Probability Letters*, *54*(4), 437–447. https://doi.org/https://doi.org/10.1016/S0167-7152(01)00124-9

Yu, K., & Zhang, J. (2005). A three-parameter asymmetric laplace distribution and its extension. *Communications in Statistics—Theory and Methods*, *34*(9-10), 1867–1879. https://doi.org/10.1080/03610920500199018

Zanini, E., Eastoe, E., Jones, M. J., Randell, D., & Jonathan, P. (2020). Flexible covariate representations for extremes. *Environmetrics*, *31*, e2624. https://doi.org/10.1002/env.2624

Appendix A

A.1 Derivation of a Degenerate Normal Distribution for P-Splines

This appendix provides a formal derivation of the degenerate Multivariate Normal distribution that arises from the Bayesian P-spline prior with a rank-deficient precision matrix, as introduced in Chapter 4.

Adapting the characterization of a degenerate multivariate normal distribution in Fahrmeir et al. (2013, pp. 650–651) to the Bayesian P-spline context, it follows:

Let $\vec{\gamma} \sim N(\vec{0}, \frac{1}{\lambda^2}\mathbf{K})$, where $\frac{1}{\lambda^2}\mathbf{K}$ with $\lambda^2 \in \mathbb{R}$ and $\in \mathbb{R}^{D\times D}$ is a precision matrix with $\text{rank}(\mathbf{K}) = r < D$. Assume that $(\mathbf{G}\,\mathbf{H})$ is an orthogonal matrix, where the columns of the $(D \times r)$-matrix $\mathbf{G}$ form a basis for the column space of $\mathbf{K}$ and the columns of $\mathbf{H}$ form a basis of the null space of $\mathbf{K}$. Consider the transformation

$$\begin{pmatrix}\vec{B}_1\\ \vec{B}_2\end{pmatrix} = \begin{pmatrix}\mathbf{G}^T\\ \mathbf{H}^T\end{pmatrix}\vec{\gamma} = \begin{pmatrix}\mathbf{G}^T\vec{\gamma}\\ \mathbf{H}^T\vec{\gamma}\end{pmatrix} \tag{A.1}$$

It follows that $\vec{B}_1$ is the stochastic part of $\vec{\gamma}$ with

$$\vec{B}_1 \sim N(\mathbf{G}^T\vec{0}, \frac{1}{\lambda^2}(\mathbf{G}^T\mathbf{K}\mathbf{G})), \tag{A.2}$$

and $\vec{B}_2$ is the deterministic part of $\vec{\gamma}$ with

$$E(\vec{B}_2) = \mathbf{H}^T\vec{0} = \vec{0}, \quad \text{Var}(\vec{B}_2) = \mathbf{0} \tag{A.3}$$

The probability density function of the stochastic part $\vec{B}_1 = \mathbf{G}^T\vec{\gamma}$ is given by

$$f(\vec{B}_1) = \frac{1}{(2\pi\lambda^2)^{\frac{r}{2}}(\prod_{i=1}^{r}\lambda_i)^{-\frac{1}{2}}}\exp\left[-\frac{1}{2\lambda^2}(\vec{\gamma}_1)^T(\mathbf{G}^T\mathbf{K}\mathbf{G})(\vec{\gamma}_1)\right] \tag{A.4}$$

where $\lambda_1, \ldots, \lambda_r$ are the r non-zero eigenvalues of $\mathbf{K}$.

To achieve the degenerate Multivariate Normal distribution form presented in Chapter 4, this thesis performs multiple steps:

1. The precision matrix $\mathbf{K}$ can be expressed through its eigendecomposition as:

$$\mathbf{K} = \mathbf{G}\mathbf{\Lambda}\mathbf{G}^T \tag{A.5}$$

   where $\mathbf{\Lambda} \in \mathbb{R}^{r \times r}$ is a diagonal matrix containing the r non-zero eigenvalues of $\mathbf{K}$.

2. From this decomposition, the following relationship for the quadratic form in the exponent of the density is established:

$$(\mathbf{G}^T\overrightarrow{\gamma})^T(\mathbf{G}^T\mathbf{K}\mathbf{G})(\mathbf{G}^T\overrightarrow{\gamma}) = (\mathbf{G}^T\overrightarrow{\gamma})^T\mathbf{\Lambda}(\mathbf{G}^T\overrightarrow{\gamma}) \tag{A.6}$$

$$= \overrightarrow{\gamma}^T\mathbf{G}\mathbf{\Lambda}\mathbf{G}^T\overrightarrow{\gamma} \tag{A.7}$$

$$= \overrightarrow{\gamma}^T\mathbf{K}\overrightarrow{\gamma} \tag{A.8}$$

3. Using the definition of the pseudo-determinant $|\cdot|_+$, the following equality holds:

$$\prod_{i=1}^{r} \lambda_i = |\mathbf{K}|_+ \tag{A.9}$$

4. Substituting these results into the density for $\overrightarrow{B}_1$ and considering that the complete density for $\overrightarrow{\gamma}$ is determined by its stochastic part (since the deterministic part adds no variance), this yields:

$$f(\overrightarrow{\gamma}) = \frac{1}{(2\pi\lambda^2)^{\frac{r}{2}}|\mathbf{K}|_+^{-\frac{1}{2}}} \exp\left[-\frac{1}{2\lambda^2}\overrightarrow{\gamma}^T\mathbf{K}\overrightarrow{\gamma}\right] \tag{A.10}$$

5. In the context of P-splines with a difference penalty of order k, the rank deficiency is $\text{rank}(\mathbf{K}) = D - k$, giving $r = D - k$. Substituting this value results in the form used in Section 3:

$$p(\overrightarrow{\gamma} \mid \lambda^2) = \frac{1}{(2\pi\lambda^2)^{\frac{D-k}{2}}|\mathbf{K}|_+^{-\frac{1}{2}}} \exp\left[-\frac{1}{2\lambda^2}\overrightarrow{\gamma}^T\mathbf{K}\overrightarrow{\gamma}\right] \tag{A.11}$$

Appendix B

B.1 Score Gradient Estimator

The Score Gradient Estimator, also called the REINFORCE gradient estimator, allows estimation of $\text{ELBO}(\vec{\theta}, \vec{\phi})$ gradients with respect to the variational parameters $\vec{\theta}$ in $q(\vec{\theta} \mid \vec{\phi})$ (Murphy, 2023, p. 268). The derivation of the Score Gradient Estimator proceeds as follows:

$$
\begin{aligned}
\nabla_{\vec{\phi}} \text{ELBO}(\vec{\theta}, \vec{\phi}) &= \nabla_{\vec{\phi}} \int q(\vec{\theta} \mid \vec{\phi}) \ln \frac{p(\mathscr{D}, \vec{\theta})}{q(\vec{\theta} \mid \vec{\phi})} d\vec{\phi} \\
&= \int \nabla_{\vec{\phi}} q(\vec{\theta} \mid \vec{\phi}) \ln \frac{p(\mathscr{D}, \vec{\theta})}{q(\vec{\theta} \mid \vec{\phi})} d\vec{\phi} \\
&= \int \left[\nabla_{\vec{\phi}} q(\vec{\theta} \mid \vec{\phi}) \right] \ln \frac{p(\mathscr{D}, \vec{\theta})}{q(\vec{\theta} \mid \vec{\phi})} d\vec{\phi} \\
&\quad + \int q(\vec{\theta} \mid \vec{\phi}) \left[\nabla_{\vec{\phi}} \ln \frac{p(\mathscr{D}, \vec{\theta})}{q(\vec{\theta} \mid \vec{\phi})} \right] d\vec{\phi} \\
&= \int \left[\nabla_{\vec{\phi}} q(\vec{\theta} \mid \vec{\phi}) \right] \ln \frac{p(\mathscr{D}, \vec{\theta})}{q(\vec{\theta} \mid \vec{\phi})} d\vec{\phi} \\
&\quad + \int q(\vec{\theta} \mid \vec{\phi}) \left[\nabla_{\vec{\phi}} \left[\ln p(\mathscr{D}, \vec{\theta}) - \ln q(\vec{\theta} \mid \vec{\phi}) \right] \right] d\vec{\phi} \\
&= \int \left[\nabla_{\vec{\phi}} q(\vec{\theta} \mid \vec{\phi}) \right] \ln \frac{p(\mathscr{D}, \vec{\theta})}{q(\vec{\theta} \mid \vec{\phi})} d\vec{\phi} \\
&\quad + \int q(\vec{\theta} \mid \vec{\phi}) \underbrace{\nabla_{\vec{\phi}} \ln p(\mathscr{D}, \vec{\theta})}_{0} d\vec{\phi} \\
&\quad - \int q(\vec{\theta} \mid \vec{\phi}) \nabla_{\vec{\phi}} \ln q(\vec{\theta} \mid \vec{\phi}) \, d\vec{\phi}
\end{aligned}
$$

$$
\begin{aligned}
&= \int \left[\nabla_{\vec{\phi}} q(\vec{\theta} \mid \vec{\phi})\right] \ln \frac{p(\mathscr{D}, \vec{\theta})}{q(\vec{\theta} \mid \vec{\phi})} d\vec{\phi} \\
&\quad - \int q(\vec{\theta} \mid \vec{\phi}) \left[\frac{1}{q(\vec{\theta} \mid \vec{\phi})} \nabla_{\vec{\phi}} q(\vec{\theta} \mid \vec{\phi})\right] d\vec{\phi} \\
&= \int \left[\nabla_{\vec{\phi}} q(\vec{\theta} \mid \vec{\phi})\right] \ln \frac{p(\mathscr{D}, \vec{\theta})}{q(\vec{\theta} \mid \vec{\phi})} d\vec{\phi} \\
&\quad - \nabla_{\vec{\phi}} \int \frac{q(\vec{\theta} \mid \vec{\phi})}{q(\vec{\theta} \mid \vec{\phi})} q(\vec{\theta} \mid \vec{\phi}) \, d\vec{\phi} \\
&\quad \text{Note: } \nabla_{\vec{\phi}} q(\vec{\theta} \mid \vec{\phi}) = q(\vec{\theta} \mid \vec{\phi}) \nabla_{\vec{\phi}} \ln q(\vec{\theta} \mid \vec{\phi}) \\
&= \int q(\vec{\theta} \mid \vec{\phi}) \left[\nabla_{\vec{\phi}} \ln q(\vec{\theta} \mid \vec{\phi})\right] \ln \frac{p(\mathscr{D}, \vec{\theta})}{q(\vec{\theta} \mid \vec{\phi})} d\vec{\phi} \\
&\quad - 0 \\
&= \mathbb{E}_{\vec{\theta} \sim q(\vec{\theta} \mid \vec{\phi})} \left[\left[\nabla_{\vec{\phi}} \ln q(\vec{\theta} \mid \vec{\phi})\right] \ln \frac{p(\mathscr{D}, \vec{\theta})}{q(\vec{\theta} \mid \vec{\phi})}\right] \\
&\overset{MC}{\approx} \frac{1}{S} \sum_{s=1}^{S} \left[\nabla_{\vec{\phi}} \ln q(\vec{\theta} \mid \vec{\phi})\right] \ln \frac{p(\mathscr{D}, \vec{\theta})}{q(\vec{\theta} \mid \vec{\phi})} \qquad \text{(B.1)} \\
&\quad \text{with } \vec{\theta}^{\,s} \sim q(\vec{\theta} \mid \vec{\phi}).
\end{aligned}
$$

Consequently, the gradient of the ELBO with respect to the variational parameters $\vec{\phi}$ can be expressed as an expected value over the latent variables $\vec{\theta}$ distributed according to the variational distribution $q(\vec{\theta} \mid \vec{\phi})$. The expected value, and thus the gradient, can be approximated by Monte Carlo integration (Ranganath et al., 2014; Murphy, 2023, (pp. 477–478)): (1) obtain S samples of $\vec{\theta}$ by sampling from $q(\vec{\theta} \mid \vec{\theta})$, (2) compute the gradient for each sample, and (3) calculate the sample mean over all S samples. For this approximation to be possible, four requirements must be met: (1) the variational distribution $q(\cdot)$ can be evaluated, (2) the variational distribution $q(\cdot)$ is differentiable, (3) sampling from the variational distribution $q(\cdot)$ is computationally efficient, and (4) the joint distribution $p(\mathscr{D}, \vec{\theta})$ can be evaluated (Ranganath et al., 2014).

However, a key challenge with the Score Gradient Estimator is the high variance of the estimated gradients. Researchers have extended the Score Gradient Estimator to reduce its variance by using control variates (Paisley et al., 2012) and Rao-Blackwellization (Ranganath et al., 2014).

Appendix C

C.1 Stick-the-Landing Gradient Estimator

Roeder et al., 2017 expanded the reparameterization trick gradient estimator:

$$\begin{aligned}\nabla_{\vec{\phi}}\,\mathrm{ELBO}(\vec{\theta},\vec{\phi}) &\overset{MC}{\approx} \nabla_{\vec{\phi}}\,\frac{1}{S}\sum_{s=1}^{S}\ln\frac{p(\mathscr{D},\vec{\theta}^{\,s})}{q(\vec{\theta}^{\,s}|\vec{\phi})}\\ &= \frac{1}{S}\sum_{s=1}^{S}\nabla_{\vec{\phi}}\ln\frac{p(\mathscr{D},\vec{\theta}^{\,s})}{q(\vec{\theta}^{\,s}|\vec{\phi})}\end{aligned} \tag{C.1}$$

For better understanding:

1. $p(\mathscr{D},\vec{\theta}^{\,s}) = f(\vec{\theta}^{\,s})$ and $\ln[f(\vec{\theta}^{\,s})] = h(\vec{\theta}^{\,s})$

2. $q(\vec{\theta}^{\,s}|\vec{\phi}) = j(\vec{\theta}^{\,s})$ and $\ln[j(\vec{\theta}^{\,s})] = m(\vec{\theta}^{\,s}) = m(g_{\vec{\phi}}(\vec{\epsilon}))$

Also note that the following holds:

$$\frac{d}{dx}r(x,l(x)) = \frac{\partial r}{\partial x} + \frac{\partial r}{\partial l}\cdot\frac{dl}{dx}$$

$$= \frac{1}{S}\sum_{s=1}^{S}\nabla_{\vec{\phi}}\frac{h(\vec{\theta}^{\,s})}{m(\vec{\theta}^{\,s})}$$

$$= \frac{1}{S}\sum_{s=1}^{S}\left[\nabla_{\vec{\phi}}\,h(\vec{\theta}^{\,s}) - \nabla_{\vec{\phi}}\,m(\vec{\theta}^{\,s})\right]$$

$$\begin{aligned}= \frac{1}{S}\sum_{s=1}^{S}\Big[&\nabla_{\vec{\phi}}\,h(\vec{\theta})\big|_{\vec{\theta}=\vec{\theta}^{\,s}}\cdot\nabla_{\vec{\phi}}\,g_{\vec{\phi}}(\vec{\epsilon})\\ &-\nabla_{\vec{\phi}}\,m(\vec{\theta})\big|_{\vec{\theta}=\vec{\theta}^{\,s}} - \nabla_{\vec{\phi}}\,m(\vec{\theta})\big|_{\vec{\theta}=\vec{\theta}^{\,s}}\cdot\nabla_{\vec{\phi}}\,g_{\vec{\phi}}(\vec{\epsilon})\Big]\end{aligned}$$

$$= \frac{1}{S}\sum_{s=1}^{S}\Big[(\nabla_{\vec{\phi}}\,(h(\vec{\theta}) - m(\vec{\theta}))\big|_{\vec{\theta}=\vec{\theta}^{\,s}}) \cdot \nabla_{\vec{\phi}}\, g_{\vec{\phi}}(\vec{\epsilon}) - \nabla_{\vec{\phi}}\, m(\vec{\theta})\big|_{\vec{\theta}=\vec{\theta}^{\,s}}\Big]$$

Resubstituting original expressions:

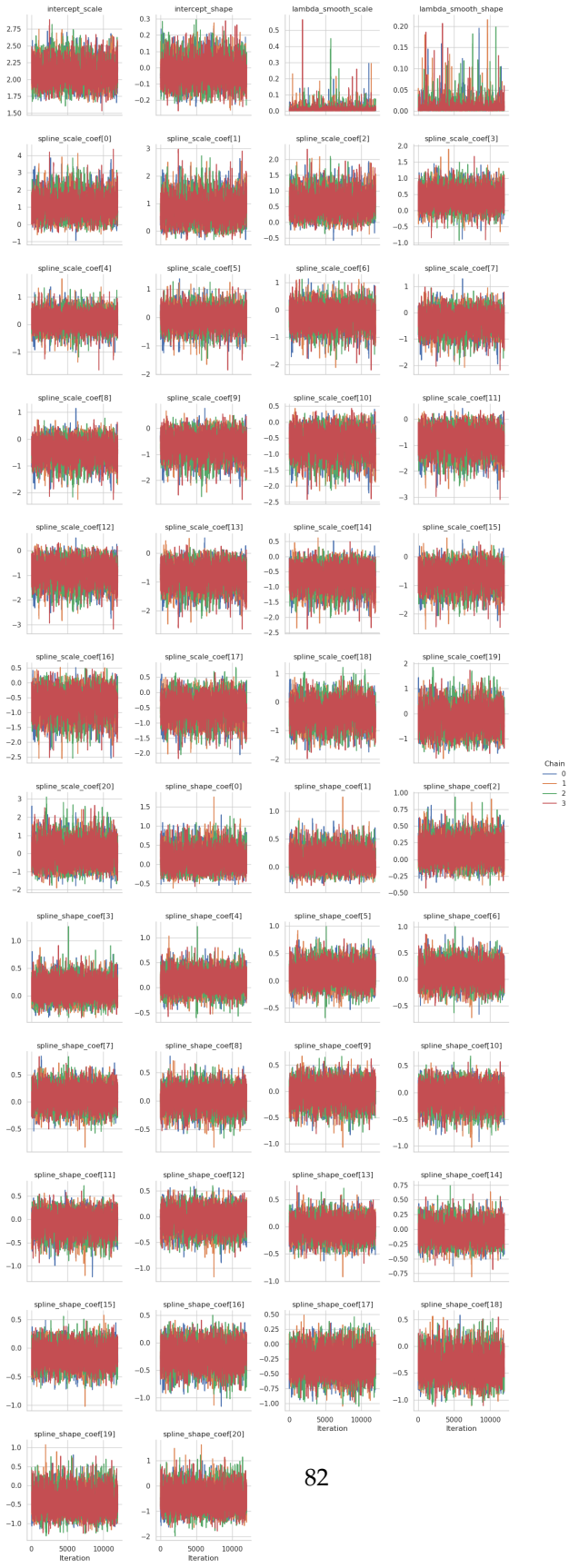
$$= \frac{1}{S}\sum_{s=1}^{S}\Big[(\nabla_{\vec{\phi}}\,(\ln p(\mathscr{D}, \vec{\theta}) - \ln q(\vec{\theta}\,|\,\vec{\phi}))\big|_{\vec{\theta}=\vec{\theta}^{\,s}}) \cdot \nabla_{\vec{\phi}}\, g_{\vec{\phi}}(\vec{\epsilon}) - \nabla_{\vec{\phi}} \ln q(\vec{\theta}\,|\,\vec{\phi})\big|_{\vec{\theta}=\vec{\theta}^{\,s}}\Big]$$

$$= \frac{1}{S}\sum_{s=1}^{S}\Big[\underbrace{\nabla_{\vec{\phi}} \ln \frac{p(\mathscr{D}, \vec{\theta})}{q(\vec{\theta}\,|\,\vec{\phi})}\Big|_{\vec{\theta}=\vec{\theta}^{\,s}} \cdot \nabla_{\vec{\phi}}\, g_{\vec{\phi}}(\vec{\epsilon})}_{\text{path derivative}} - \underbrace{\nabla_{\vec{\phi}} \ln q(\vec{\theta}\,|\,\vec{\phi})\big|_{\vec{\theta}=\vec{\theta}^{\,s}}}_{\text{score function}}\Big], \tag{C.2}$$

and found that for some models, computing the gradient only for the path derivative term while eliminating the score function term leads to lower variance and faster convergence of the Reparameterization Gradient Estimator.

Appendix D

D.1 Simulation Study MCMC Baseline Chain Plots



82

Figure D.1: Chains for $N = 250$ exhibit good mixing.

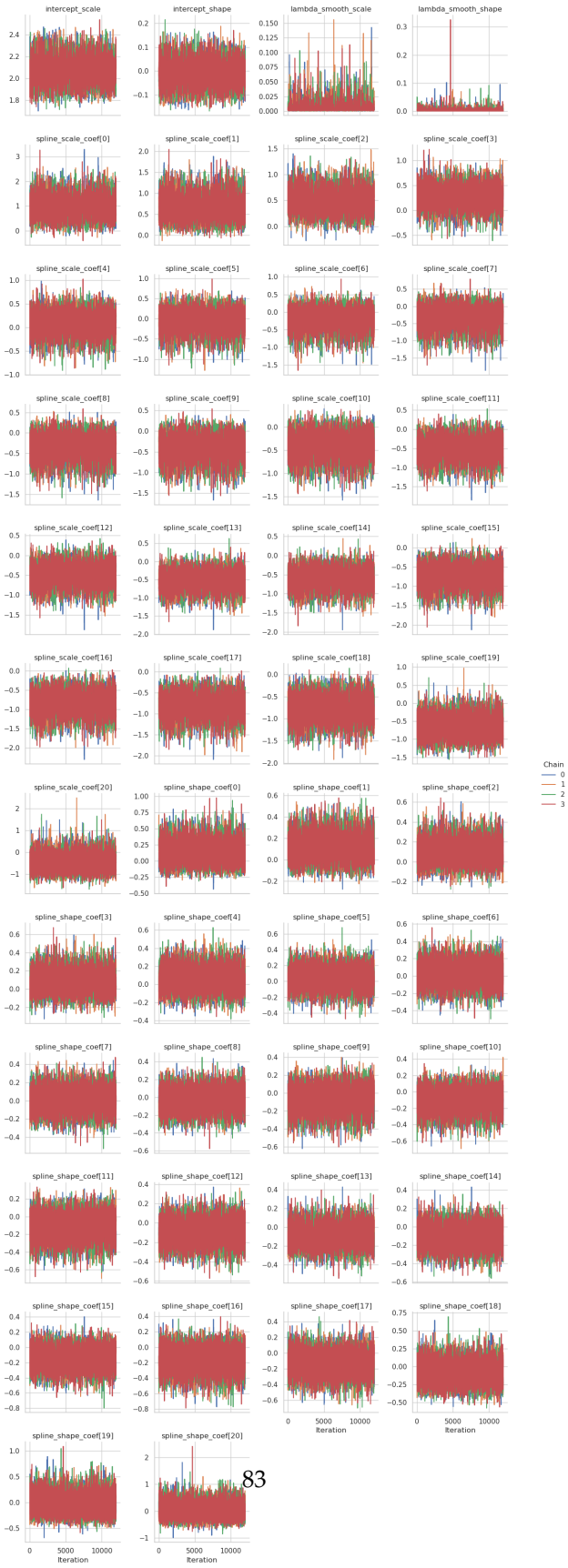


Figure D.2: Chains for $N = 500$ exhibit good mixing.

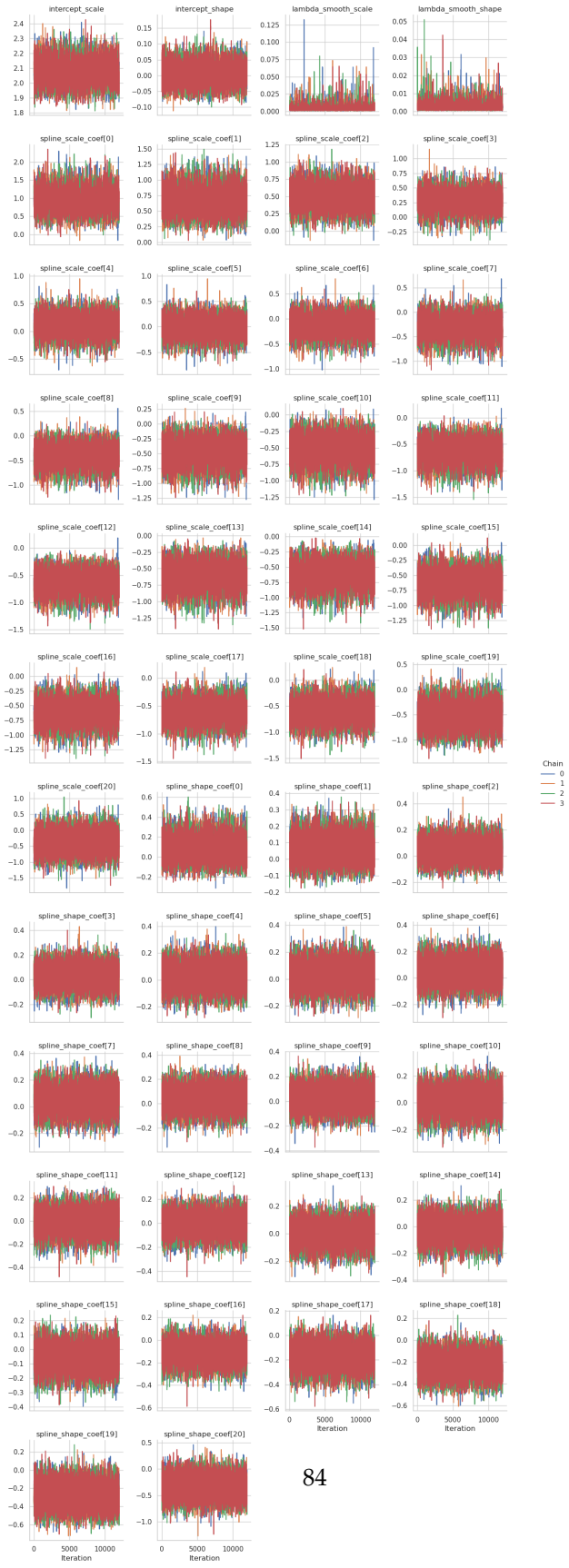


Figure D.3: Chains for $N = 1000$ exhibit good mixing.

Appendix E

E.1 Case Study MCMC Chain Plot

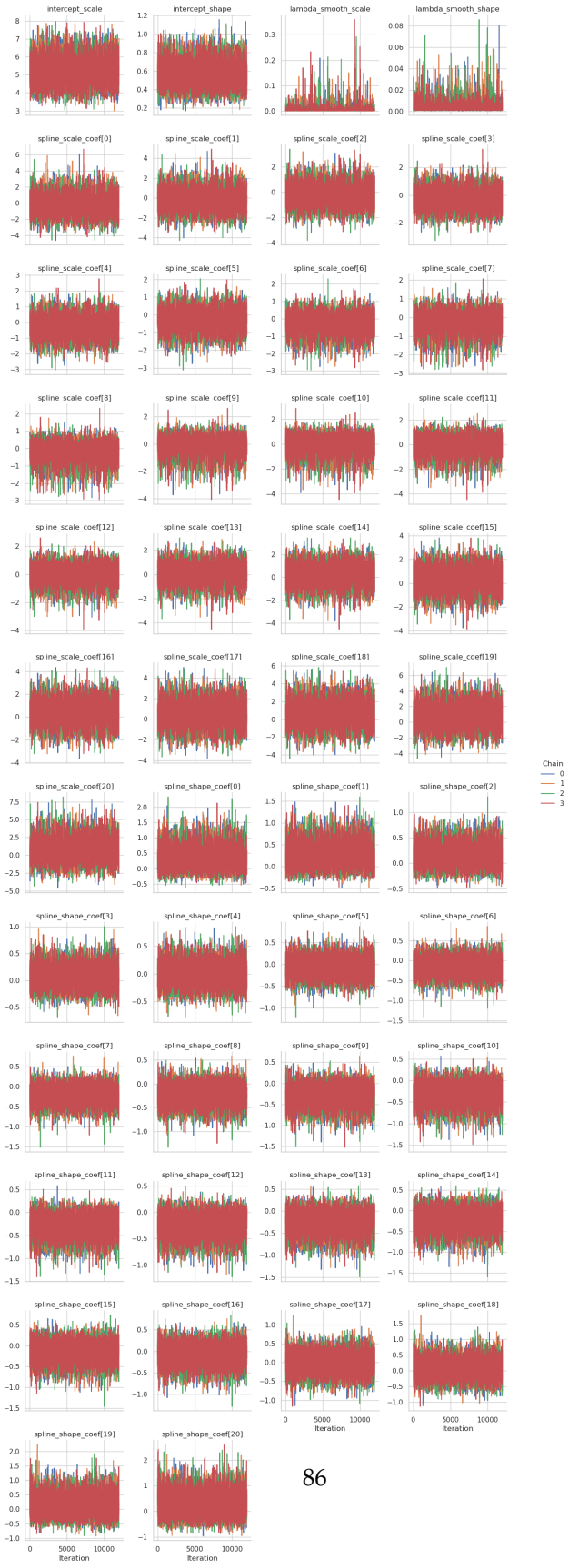


Figure E.1: Chains for Danish Fire Insurance Exceedances over Threshold exhibit good mixing.