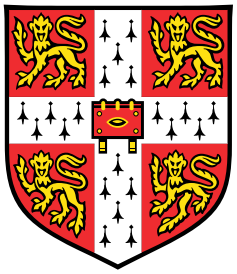
Non-Gaussian Lévy Processes in Machine Learning



Trevor Clark

Machine Learning and Machine Intelligence
Department of Engineering
University of Cambridge

This dissertation is submitted for the degree of
Master of Philosophy

Fitzwilliam College August 2022

Declaration

I, Trevor Clark of Fitzwilliam College, being a candidate for the MPhil in Machine Learning and Machine Intelligence, hereby declare that this report and the work described in it are my own work, unaided except as may be specified below, and that the report does not contain material that has already been used to any substantial extent for a comparable purpose.

Signed. Trevor Clark
Date 18 August, 2022.

For the project, I used standard scientific computing packages in python *e.g.* numpy and scipy. For working with Gaussian processes I used the package stheno https://github.com/wesselb/stheno. The starting point for the project is the the code at https://github.com/yamankindap/GiG/tree/main/python, which I reimplemented to include some improvements to the algorithms that were made in Kındap and Godsill (2022a). This is used extensively in the project for simulations and inference.

This thesis has 11225 words.

Trevor Clark
August 2022

Abstract

The chief object of study in this thesis are linear time-invariant systems that are driven by non-Gaussian noise; specifically,

$$f(t) = \int_{\mathbb{R}} h(t-\tau) dX(\tau),$$

where $\{X(\tau) : \tau \geq 0\}$ is a generalized hyperbolic (GH) Lévy process and h is a Gaussian process. These are modifications of the Gaussian Process Convolution Model which was introduced in Tobar et al. (2015) where the Lévy process was assumed to be the Wiener process. We explain the methods of Kındap and Godsill (2022a) and Godsill and Kındap (2021) used to sample (GH) Lévy processes, from this we obtain a method for sampling Lévy driven GPCMs and finally we develop an MCMC method for performing inference for these models.

To Evgenia and Matthew.

Acknowledgements

I would like to thank my advisors Simon Godsill and Yaman Kındap for their advice, time and many fruitful conversations.

Table of contents

List of figures

List of tables

Chapter 1

Introduction

The first example of a stochastic process that one usually encounters is Brownian motion. Brownian motion may be thought of as a continuous-time limit of a random walk on the real line as the step-size tends to zero, and it is used to model the random motion of a particle suspended in a liquid or gas. One of its defining characteristics is that its increments $X(t)-X(s)$ are normally distributed. Specifically for the standard Brownian motion

$$X(t)-X(s) \sim \mathcal{N}(0,t-s).$$

Thus we see that there is an intimate connection between Brownian motion and the Gaussian distribution.

While the Gaussian distribution fits many natural phenomena, there are numerous real-world datasets which exhibit "heavy-tails" and assuming that such data is Gaussian is not appropriate. Indeed, there are many examples of random processes whose increments should be modelled by a non-Gaussian distribution. Examples arise in finance, Cont and Tankov (2003), Mandelbrot (1963); signal processing, Nikias and Shao (1995); climate science, Katz and Brown (1992) and elsewhere. The introduction of Godsill and Kındap (2021) gives a thorough outline of applications of non-Gaussian processes. When working with non-Gaussian processes or distributions, one often sacrifices the tractability of the Gaussian; in particular, sampling from these distributions can be difficult, and only recently have methods been developed to sample the paths of certain non-Gaussian stochastic processes, Godsill and Kındap (2021); Kındap and Godsill (2022a).

In this thesis, we will be concerned with stochastic processes $\{X(t) : t \geq 0\}$ that are *stationary*, so that for any $t > s \geq 0$ in the domain of the process, we have that the increment $X(t)-X(s)$ has the same distribution as $X(t-s)$, and processes where the distributions of the values at time t comes from a specified family of distributions. The family of non-Gaussian

distributions that is most relevant for us are the generalized hyperbolic (GH) distributions. Generalized hyperbolic distributions were introduced in Barndorff-Nielsen and Halgreen (1977), where they were used to model the physics of wind-blown sand. More recently, they have been applied in financial modelling, for example in Eberlein (2001). The family of GH distributions contains several important subclasses *e.g.* the hyperbolic, the normal inverse Gaussian and the Student's t-distributions. We will discuss some of the properties of this distribution that are relevant for us in Section 2.2.

In Barndorff-Nielsen and Halgreen (1977) it is shown that a GH distribution is *infinitely divisible*, see page 12, which implies that any GH distribution arises as the distribution of the values of a stochastic process at time one. We describe the recent methods of Godsill and Kındap (2021) and Kındap and Godsill (2022a) used to simulate these processes in Chapter 3. We will see that such processes are pure jump Lévy processes (see Section 2.3), which can be represented by an infinite sum of jumps and can be approximated well by a process determined by a finite sum of jumps at uniformly distributed jump times. Generalized hyperbolic processes have almost surely infinitely many jumps in every time interval of positive length, so in simulations, some approximation of them is always necessary. We will see that there is a principled and effective method for choosing the finitely many jumps used to approximate a GH process.

The feature of being able to approximate a GH process well by a finite number of jumps makes them amenable to simulation and also to inference, as we will see. For example, using this approximation of a GH process one can easily simulate Ornstein-Uhlenbeck systems (and even more general CARMA systems) that are driven by generalized hyperbolic Lévy noise.

The principle object that we study in this thesis are linear time-invariant systems driven by such Lévy processes. Tobar et al. (2015) introduced the Gaussian Process Convolution Model (GPCM). They modelled signals as the convolution of a Gaussian process h with white noise x; this model may also be expressed as the stochastic integral of h with respect to the Wiener process $\{W(\tau) : \tau \geq 0\}$:

$$f(t) = \int_{\mathbb{R}} h(t-\tau)x(\tau)d\tau = \int_{\mathbb{R}} h(t-\tau)dW(\tau).$$

We will consider models that are obtained by replacing the Wiener process with a GH Lévy process. In this case we have that

$$f(t) = \int_{\mathbb{R}} h(t-\tau)dX(\tau) = \sum_{j=1}^{\infty} h(t-\tau_j)dX_{\tau_j},$$

where the τ_j are the jump times of the Lévy process and dX_{τ_j} is the size of the jump at time τ_i. Since we approximate our processes by a finite number of jumps, the sum on the right becomes a finite sum. Combining the methods of Godsill and Kındap (2021); Kındap and Godsill (2022a) with sampling of Gaussian processes, we have an effective method for sampling such objects. In Chapter 4, we will describe and evaluate the use of Metropolis-Hastings based methods for performing inference using these models.

1.1 Stochastic processes

Let us now discuss stochastic processes a little more formally, and introduce the important special case of Gaussian processes.

A *stochastic process* is a set of random variables indexed by a set. We will only consider continuous time processes indexed by intervals $I \subset \mathbb{R}$, the real-line, taking values in $\mathbb{R}$. For example, the one-dimensional *Brownian motion*, also known as the *Wiener process*, $\{X_t : t \geq 0\}$, is characterized by the following conditions (Bass, 2011):

- The intial value $X_0 = 0$.
- *Independence of increments*: for any sequence of times $0 \leq t_0 < t_1 < t_2 < \cdots < t_{n-1} < t_n$, the values of the increments $X_{t_i} - X_{t_{i-1}}$, $1 \leq i \leq n$, are mutually independent.
- For any $0 \leq s < t$, the distribution $X_t - X_s$ is normally distributed with mean zero and variance $t - s$.
- X_t is continuous in t.

We show some sample paths of the Brownian motion in Figure 1.1.

We will discuss more general Lévy processes in more detail in Section 2.3, but let us point out that they address some important considerations when modelling certain real-world data. General Lévy processes are not required to be continuous. This makes them suitable for real-world applications where one may observe jumps in the data. Also, observe that in the definition of Brownian motion, we assume that the increments $X_t - X_s$ for $s < t$ are Gaussian. We illustrate this in Figure 1.2. For general Lévy processes, this condition is relaxed, and one only requires *stationary increments*, that the distribution of $X_t - X_s$ is the same as the distribution of X_{t-s}. More specifically, a generalized hyperbolic process has the property that the distribution of its values at time one is a generalized hyperbolic distribution, see Section 2.2.

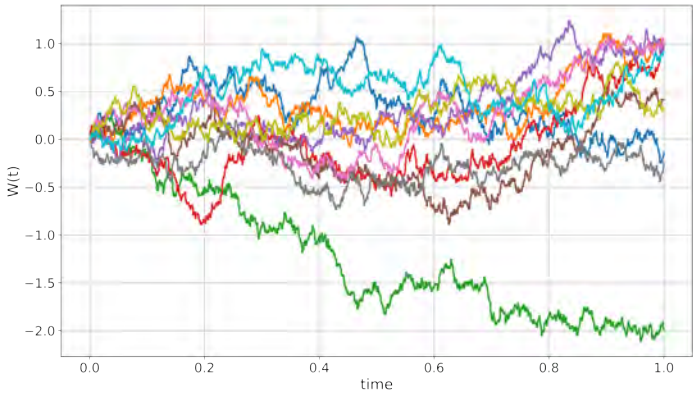


Fig. 1.1 Sample paths of Brownian motion on the real-line.

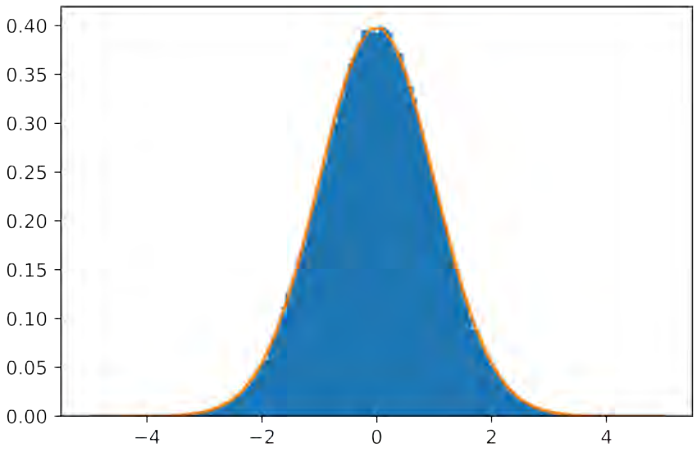


Fig. 1.2 Histogram of time-one values of the Brownian motion on $\mathbb{R}$, the Wiener process, (in blue) and a graph of the standard Gaussian pdf (in orange).

1.1.1 Gaussian processes

Gaussian processes are stochastic processs, which generalize the Gaussian distribution, and they provide us with a powerful tool for modelling unknown functions. Indeed they are a standard tool in Bayesian signal processing. Often they are state of the art for continuous regression problems, they provide estimates for the uncertainty of their predictive distributions, and are they are interpretable. In this section, we provide some relevant background material on Gaussian processes, and we refer to Rasmussen and Williams (2006) for the details and further information.

A Gaussian process is a stochastic process that is specified by a mean function $\mu(\cdot)$ and a positive definite *covariance function*, also called a *kernel*, $\mathscr{K}(\cdot,\cdot)$. This defines a distribution over functions with the property that for any finite vector of points $x = (x_1,\ldots,x_n)$, we have that the joint distribution $p(f(x)) = \mathscr{N}(f(x);\mu(x),\mathscr{K}(x,x))$, where $f(x) = (f(x_1),\ldots,f(x_n))$, $\mu(x) = (\mu(x_1),\ldots,\mu(x_n))$, and the covariance matrix $\mathscr{K}(x,x)$ has i,j entry $\mathscr{K}(x_i,x_j)]$ for $1 \leq i,j \leq n$. We will write $f(x) \sim GP(\mu(x),\mathscr{K}(x,x))$.

An important feature of Gaussian processes, and one which motivated the development of the GPCM is that Gaussian processes provide a non-parametric model for an unknown function - a Gaussian process cannot be parameterized by finite set of parameters. This aspect of Gaussian processes allows them to model complex functions when they are provided with enough data, while being robust against over-fitting when little data is available.

An example of a covariance function is the *exponential quadratic* kernel

$$\mathscr{K}_{EQ}(x,x') = e^{-\frac{1}{2}\|x-x'\|^2}.$$

This kernel is stationary. Also, it corresponds to a local smoothing operation and any $f(x) \sim GP(\mu(x),\mathscr{K}_{EQ}(x,x))$ is smooth. So we see that simple choices of kernel, such as this one, limit the ability of the Gaussian process to generalize.

The first step in Gaussian process regression is the specification of a prior distribution over functions, which captures underlying assumptions about the target function, for example its smoothness, periodicity or whether it is stationary. These assumptions about the function guide the choice of the prior distribution, and hence the kernel. The chief modelling decision that is made when using Gaussian processes is arguably the choice of kernel, Tobar et al. (2015). Finding methods for constructing kernels as well as determining the best kernel for a given task are both active research areas.

1.2 The Gaussian Process Convolution Model

Finding principled methods for choosing which kernel to use when modelling with a Gaussian process and constructing expressive kernels are important problems, Tobar et al. (2015), and these are active areas of research. Three approaches to this problem have been pursued. One approach is to start with a collection of basic kernels, and search through the space of kernels for an effective one for a particular problem by composing the basic examples, Grosse et al. (2012), Duvenaud et al. (2013) and Malkomes et al. (2016). A second, is to obtain a parametrized family of kernels by parametrizing the power spectral density of the kernel using Gaussians, Wilson and Adams (2013), Lévy processes, Jang et al. (2017), or Dirichlet processes, Oliva et al. (2016). Other methods for parametrizing the kernel were used in Calandra et al. (2016) and Sun et al. (2018). The third approach, and the one that is motivates part of this work, is to treat the kernels non-parametrically (in the same way that Gaussian processes themselves treat functions non-parametrically). This attempts to resolve problems of tractability in the first approach and over-fitting in the second. This idea was developed in Tobar et al. (2015) and Bruinsma et al. (2022). Let us briefly describe the GPCM model of Tobar et al. (2015), and how it relates the problem of finding an appropriate kernel. In Tobar et al. (2015), a signal is modelled as the a linear, time-invariant system obtained as the stochastic integral of a Gaussian process with respect to the Wiener process. Now, given a filter $h \sim GP(\mu, \mathscr{K})$, the process

$$f(t) = \int_{\mathbb{R}} h(t-\tau) dW(\tau),$$

is again a Gaussian process, since $f(t)$ is a linear combination of Gaussian random variables, and, since h is drawn from a non-parametric prior, we obtain a non-parametric prior over kernels. For this purpose there is nothing special about the Wiener process, and replacing it with a general Lévy process, provides us with a means to construct more general priors over kernel functions.

In Tobar et al. (2015), the filter is chosen from a Gaussian process with the *decaying exponential quadratic kernel*:

$$\mathscr{K}_{DEQ}(t_1, t_2) = \sigma_h e^{-\alpha(t_1^2+t_2^2)-\gamma(t_1-t_2)^2},$$

where σ_h $w = 1/\sqrt{\alpha}$ is the *window* parameter, $s = 1/\sqrt{\gamma}$ is the *length-scale* parameter and σ_h is the *strength*. As for the exponential quadratic kernel, the length-scale parameter determines the time-scale over which filter varies and the window parameter determines the extent of the

domain over which the filter is active, which in turn determines the length of time correlations in the output signal Tobar et al. (2015).

These models are incredibly flexible, and there are numerous ways in which they can be modified. In Bruinsma et al. (2022) two modification of this model were introduced. One is the Causal Gaussian Process Convolution Model CGPCM. This model is a modification of the GPCM, so that as is the case with physical systems, the values of the output f do not depend on future values of the input. For the CGPCM given $h \sim GP(0, \mathscr{K}_{DEQ})$, we have

$$f(t) = \int_{-\infty}^{t} (h(t-\tau))sW(\tau).$$

It is proved in Bruinsma et al. (2022) that if $h(0) \neq 0$, then sample paths of the CGPCM are almost surely nowhere differentiable.

A second variation of the GPCM that was introduced in Bruinsma et al. (2022) is the Rough Gaussian Process Convolution Model, RGPCM. The RGPCM is defined by the following generative model. Let h be white noise windowed by $e^{-\alpha|t|}$, that is, $h \sim GP(0, k_h)$, where $k_h(t_1,t_2) = \tilde{\alpha}^2 e^{-\alpha|t|-\alpha|t'|}\delta(t_1-t_2)$, where δ is the Dirac-δ function, and we take $x \sim GP(0,K)$, where $K(t_1,t_2) = e^{-\lambda|t_1-t_2|}$ is the the Matèrn$-1/2$ kernel. Then

$$f(t) = \int_{-\infty}^{t} h(t-\tau)x(\tau)\,d\tau.$$

The RGPCM can model sample paths that are more irregular than the CGPCM Bruinsma et al. (2022).

We investigate (acausal) GPCM with the DEQ-kernel, but which are driven by Lévy noise. The sample of these models are smooth; however they are able to model jumps in functions that may be difficult to model when the input to the system is Brownian motion.

1.3 Starting points for the project

This project built on code that was made available by the advisors for the project, Godsill and Kindap, at https://github.com/yamankindap/GiG. I re-implemented this with some small changes to incorporate some of the improvements to the sampling procedure that were introduced in Kındap and Godsill (2022a). This code underlies the entire project, which depends on sampling paths from generalized inverse Gaussian distributions.

To work with Gaussian processes, I made used of the stheno package https://github.com/wesselb/stheno. This plays an important role in Chapter 4, where we need to sample from Gaussian processes as part of the Gibbs sampler in our inference scheme. I also investigated

the implementation of the its pseudopoints approximation of Gaussian processes; however, this is not crucial for the project.

Otherwise, the code for this project was written using standard scientific computing packages in python: numpy, scipy and jax.numpy to parallelize a few computations.

1.4 Outline

- In Chapter 2, we provide background material for the thesis. We start by presenting a few key details from probability theory and go on to give a brief outline of the theory of Lévy processes, which feature in Chapters 3 and 4. We also give a brief description the Metropolis-Hastings Algorithm and Gibbs sampling, which we will need in Chapter 4.

- In Chapter 3, we describe the methods of Godsill and Kındap (2021); Kındap and Godsill (2022a) for generating sample paths of GH processes and present simulations of GH processes.

- Chapter 4 develops tools for inference for certain linear time invariant systems that are driven by Gaussian noise. We make use of a Gibbs sampler. Fortunately, some of our marginals we need to sample from are Gaussians, which are straight-forward to sample from once we compute their means and covariances. However, we also need to sample paths of a Lévy process which we do by implementing a Metropolis-Hastings within Gibbs algorithm. We go on to evaluate the sampler, and discuss some improvements to it.

- In Chapter 5, we evaluate the sampler and the Lévy driven GPCM on the price of crude oil, and compare this model to the different GPCMs of Bruinsma et al. (2022); Tobar et al. (2015).

- In Chapter 6, We present a discussion of our work and possible future directions.

Chapter 2

Background

In this chapter, we present some of the required background material we will need in Chapters 3 and 4. We begin this section with a brief discussion of generalized hyperbolic and generalized inverse Gaussian distributions and we go on to discuss Lévy processes and the Gaussian process convolution model. We conclude with a description of the Metropolis-Hastings algorithm.

2.1 Characteristic functions

Characteristic functions will serve as an indispensable tool later for understanding the structure of Lévy processes. Characteristic functions are Fourier transforms of probability measures: Let X be a real-valued random variable defined on a probability space with distribution p. The *characteristic function of* X, $\phi_X : \mathbb{R} \to \mathbb{C}$, from the real line to complex plane, is defined by $\phi_X(u) = \mathbb{E}(e^{iuX}) = \int e^{iuy} p(dy)$. We also call this object the *characteristic function of* p, and write $\phi_p = \phi_X$, as the expected value does not depend on X.

Two important examples are

- the characteristic function of a Gaussian $p(x) = \mathcal{N}(x; \mu, \sigma^2)$ is given by

$$\phi_p(u) = e^{i\mu u - \frac{1}{2}\sigma^2 u^2} \quad \text{and}$$

- the characteristic function of a Poisson distribution $\psi(x) = \text{Pois}(x; \lambda)$ is

$$\phi_\psi(u) = \exp\left[\lambda(e^{iu} - 1)\right].$$

There is a bijection between characteristic functions and probability distributions, so the characteristic function of a distribution completely defines it.

2.2 Generalized hyperbolic and generalized inverse Gaussian distributions

The GH and GIG families of distributions are closely related and they both play an important role in our study of GH processes.

Generalized hyperbolic distributions

The general form of the density function for a GH distribution is

$$p_{GH}(x) = a(\lambda,\alpha,\beta,\delta)\cdot(\delta^2+(x-\mu)^2)^{(\lambda-1/2)/2}\cdot K_{\lambda-1/2}\Big(\alpha\sqrt{\delta^2+(x-\mu)^2}\Big)e^{\beta(x-\mu)},$$

where

$$a(\lambda,\alpha,\beta,\delta) = \frac{(\alpha^2-\beta^2)^{\lambda/2}}{\sqrt{2\pi}\alpha^{\lambda-1/2}\delta^{\lambda}K_{\lambda}(\delta\sqrt{\alpha^2-\beta^2})},$$

and the function $K_\nu(\cdot)$ is the modified Bessel function of the second kind. It has an integral representation as

$$K_\nu(z) = \frac{1}{2}\int_0^\infty y^{\nu-1}e^{-\frac{z}{2}\left(y+\frac{1}{y}\right)}\,dy.$$

This family of distributions has five parameters:

- $\alpha > 0$ determines the shape;
- $\beta \in (0,|\alpha|)$, the skewness;
- $\mu \in \mathbb{R}$, the location;
- $\delta > 0$, the scale;
- and $\lambda \in \mathbb{R}$, the heaviness of the tails.

Generalized inverse Gaussian distributions

A related family of distributions are the generalized inverse Gaussian (GIG) distributions. The density function of a GIG distribution is given by

$$p_{GIG}(x;\delta,\gamma,\lambda) = \left(\frac{\gamma}{\delta}\right)^{\lambda}\frac{1}{2K_\lambda(\delta\gamma)}x^{\lambda-1}e^{-\frac{1}{2}(\delta^2x^{-1}+\gamma^2x)}\chi_{x>0},$$

and the parameter space is given by Eberlein (2001):

- $\delta \geq 0, \gamma > 0$, if $\lambda > 0$;
- $\delta > 0, \gamma > 0$, if $\lambda = 0$;
- $\delta > 0, \gamma \geq 0$, if $\lambda < 0$.

Special cases of the GIG distribution include the gamma, the inverse gamma and the inverse Gaussian distributions. A generalized hyperbolic distribution may represented as a mean-variance mixture of Gaussians. Indeed,

$$p_{GH}(x) = \int_{(0,\infty)} \mathcal{N}(x; \mu + \beta u, u) p_{GIG}(u; \delta, \sqrt{\alpha^2 - \beta^2}, \lambda)\, du.$$

This establishes a first relationship between GIG and GH distributions, which plays an important role in what follows.

2.3 Lévy processes

In this section, we collect relevant definitions and results concerning Lévy processes. Any of the books, Applebaum (2009); Bertoin (1996); Sato (1999), would serve as a reference for the material presented here. Much of the theory we discuss here holds for higher dimensional Lévy processes; however, for simplicity, and because it is all that is relevant for the later sections, we restrict our attention to the one-dimensional case.

A stochastic process $\{X(t) : t \geq 0\}$ is called a *Lévy process* if it satisfies the following:

- $X(0) = 0$ almost surely.
- *Independence of increments*: for any sequence of times $0 \leq t_0 < t_1 < t_2 < \cdots < t_{n-1} < t_n$, the values of the increments $X(t_i) - X(t_{i-1})$, $1 \leq i \leq n$, are mutually independent.
- *Stationary increments*: For any $0 \leq s < t$, the distribution $X(t) - X(s)$ is equal to $X(t-s)$.
- *Continuity in probability.* For any $\varepsilon > 0$ and $t \geq 0$, we have that

$$\lim_{\eta \to 0} P(|X(t+\eta) - X(t)| > \varepsilon) = 0.$$

We recall that events $A_i, i = 1, \ldots, n$ are said to be *mutually independent* if for every subset $\{i_1, \ldots, i_k\}$ of $\{1, 2, \ldots, n\}$, we have that $p(A_{i_1} \cap A_{i_2} \cap \cdots \cap A_{i_k}) = p(A_{i_1}) p(A_{i_2}) \ldots p(A_{i_k})$.

Any Lévy process $X = \{X(t) : t \geq 0\}$ has a *modification* which is continuous on the right for each $t \geq 0$ and has left limits for each $t > 0$. We call such a process *cádlág*. More precisely,

for any Lévy process $\{X(t) : t \geq 0\}$, there exists a Lévy process $X' = \{X'(t) : t \geq 0\}$ so that X' is cádlág, and $P[X_t = X'_t] = 1$ for all $t > 0$. From now on, we will assume that all Lévy processes are cádlág. This technical assumption about Lévy processes immediately implies that for each $\varepsilon > 0$, a Lévy process can have at most finitely many jumps with size greater than ε, which in turn gives us that a Lévy process can have at most countably many jumps.

2.3.1 Infinite divisibilty: existence of GH and GIG Lévy processes

Suppose that μ and η are real-valued probability measures. For any Borel set $A \subset \mathbb{R}$, we define the convolution of probability measures by the formula: $\mu * \eta(A) = \int_{\mathbb{R}} \mu(A-x)\eta(dx)$, where $A - x$ denotes the set $\{y - x : y \in A\}$. It is useful to observe that the convolution of two probability measures is a probability measure, and that convolution gives the probability distribution for the sum of two independent random variables: Suppose that X_1 and X_2 are independent random variables defined on a probability space $(\Omega, \mathscr{F}, P)$, with joint distribution p and marginals μ_1 and μ_2. Then

$$P(X_1 + X_2 \in A) = \mathbb{E}(\chi_A(X_1 + X_2)) = \mu_1 * \mu_2(A),$$

where χ_A denotes the indicator function of A.

We define $\mu^{*^n} = \mu * \cdots * \mu$, ($n$ times), and if there exists a measure ν, so that $\nu^{*^n} = \mu$, we call ν the n^{th} *root* of μ. We say that a density μ is *infinitely divisible*, μ has an n^{th} root for every $n \in \mathbb{N} = \{1, 2, \dots\}$.

The following theorem relates infinitely divisible distribution with Lévy processes.

Theorem. *If μ is an infinitely divisible probability measure on $\mathbb{R}$, then there exists a Lévy process $\{X(t) : t \geq 0\}$, so that the distribution of $X(1)$ is μ.*

Employing results of Grosswald (1976), infinite divisibility of the GH and GIG distributions was established in Barndorff-Nielsen and Halgreen (1977).

It is straightforward to see that the distribution of the time $t > 0$ values of a Lévy process is always infinitely divisible. For any $n \in \mathbb{N}$, one may express

$$X(t) = X(t/n) + (X(2t/n) - X(t/n)) + \cdots + (X(t) - X((n-1)t/n)),$$

and the infinite divisibility follows from the fact that the distributions of the increments of a Lévy process are stationary.

2.3.2 The Lévy-Khintchine Formula

Lévy-Khintchine Formula and the Lévy-Itô Decomposition are two closely related foundational results in the theory of Lévy processes, which serve to illuminate the basic structure of Lévy processes. They provide us with a decomposition of a Lévy process into parts that capture the distinct behaviours which are exhibited by Lévy processes, together with a means of determining which types of behavior are present in a given Lévy process through the form of its characteristic equation.

We define the *characteristic function of a stochastic process* $\{X(t) : t \geq 0\}$ to be $\phi(X(t)) = \phi_{X(t)}$, the characteristic function of the random variable $X(t)$. If a distribution μ is infinitely divisible, and ν is an n^{th} root of μ, then their characteristic functions satisfy

$$\phi_\nu^n = \phi_\mu.$$

It follows from the fact that Lévy processes are stationary that the characteristic function of an infinitely divisible distribution has no zeros and so can be expressed as $\phi_\mu(u) = \exp(-Z(u))$, $u \in \mathbb{R}$. Moreover,

$$\phi_{X(t)}(u) = \exp(-tZ(u)) \text{ where } \phi_{X(1)}(u) = \exp(-Z(u)).$$

This formula is immediate for rational t, and by taking limits it can be extended to general t. The Lévy-Khintchine Formula provides an explicit expression for the characteristic function.

We call a measure ν with no atom on $\{0\}$ satisfying $\int_{\mathbb{R}} x^2 \wedge 1 \, \nu(dx) < \infty$ a *Lévy measure*. Recall that if $a, b \in \mathbb{R}$, then $a \wedge b$ is defined to be the minimum of a and b.

For any set A, we let χ_A denote the indicator function ot A.

Theorem (Lévy-Khintchine). *A probability measure μ on $\mathbb{R}$ is infinitely divisible if there exists $a, b \in \mathbb{R}$ and a Lévy measure Q on $\mathbb{R} \setminus \{0\}$, so that for all $u \in \mathbb{R}$,*

$$\phi_\mu(u) = \exp\left\{ ibu - \frac{1}{2}au^2 + \int_{\mathbb{R}\setminus\{0\}} \left[e^{iuy} - 1 - iuy\chi_{(-1,1)\setminus\{0\}}(y)\right] \nu(dy) \right\}. \tag{2.1}$$

Conversely, any mapping of the form (2.1) *is the characteristic function of an infinitely divisible probability measure on $\mathbb{R}$.*

Since the the distribution of values of a Lévy process at a given time is always infinitely divisible, this theorem holds for all such distributions.

The particular form of the Lévy-Khintchine representation of the characteristic function of an infinitely divisible density gives us important information about the associated Lévy

process. This is beautifully explained in Applebaum (2009), and we recall his discussion here.

Case 1. (Linear motion or *drift*) Assume that $a = 0$ and that the Lévy measure ν vanishes. Then the characteristic formula reduces to $\phi_\mu(u) = e^{ibu}$. If $X(t)$ is the value of the process at time t, then from the definition of the characteristic function of X, we have that $\mathbb{E}(e^{iuX(t)}) = e^{ibut}$, so that $\mathbb{E}(X(t)) = bt$, is deterministic, and we have $X(t) = bt$ is motion in a straight line.

Case 2. (*Brownian motion with drift*) Assume that $a \neq 0$, but still assume that ν vanishes. Then we have that

$$\phi_{X(t)}(u) = e^{ibut - \frac{at}{2}u^2}.$$

This is the characteristic function of a Gaussian random variable $X(t)$ with mean tb and variance at. A processes with this characteristic function is known as Brownian motion with drift. If $b = 0$ and $a = 1$, then $\{X(t) : t \geq 0\}$ it is standard Brownian motion.

Case 3. (Brownian motion with drift, with discontinuities determined by a compound Poisson process.) Let us first consider the simplest case where ν is does not vanish. Assume that $\nu = \lambda \delta_h$, where δ_h is a Dirac measure supported on $h \neq 0$. Then

$$\phi_\mu(u) = \exp\left\{ ibu - \frac{1}{2}au^2 + \int_{\mathbb{R}\setminus\{0\}} \left[e^{iuy} - 1 - iuy\chi_{(-1,1)\setminus\{0\}}(y)\right] \lambda \delta_h(dy) \right\},$$

and setting $b' = b - \int_{\mathbb{R}\setminus\{0\}} y\chi_{(-1,1)\setminus\{0\}}(y)\lambda\delta_h(dy)$, the characteristic function takes the form:

$$\phi_\mu(u) = \exp\left\{ ib'u - \frac{1}{2}au^2 + \int_{\mathbb{R}\setminus\{0\}} \left[e^{iuy} - 1\right] \lambda \delta_h(dy) \right\}.$$

Hence,

$$X(t) = b't + \sqrt{a}B(t) + N(t),$$

where $B = \{B(t) : t \geq 0\}$ is standard Brownian motion, and $N = \{N(t) : t \geq 0\}$ is an independent process with characteristic function

$$\phi_{N(t)}(u) = \exp\left[\lambda t(e^{iuh} - 1)\right].$$

Thus, we see that N is a Poisson process with probability distribution:

$$P(N(t) = nh) = \frac{(\lambda t)^n}{n!}e^{-\lambda t};$$

it has intensity λ and takes values in the set $\{nz : n \in \mathbb{N}\}$.

The paths of this process follow a Brownian motion with drift up until some random time T_1 coinciding with the first jump time of the Poisson process, where it has a discontinuity of size $|h|$, and then it continues to follow a Brownian motion with drift up until some random time T_2, the second jump of the Poisson process, where again it has a discontiuity of size $|h|$, and so on.

This discussion extends immediately to case where $\nu = \sum_{i=1}^{M} \lambda_i \delta_{h_i}$ is a finite linear combination of δ functions with each $\lambda_i > 0$. In this case we have that

$$X(t) = b't + \sqrt{a}B(t) + N_1(t) + \cdots + N_M(t),$$

where $N_1, \ldots, N_M$ are independent Poisson processes (whose sum is known as a *compound Poisson process*), and N_i takes value in the set $\{nh_i : n \in \mathbb{N}\}$, so that again we have that $X(t)$ consists of segments where it follows a Brownian motion with drift, bounded by jump discontinuities, where the sizes of the jumps are from $\{h_1, \ldots, h_M\}$.

So far we have seen that when $b \neq 0$, the Lévy process possess a drift part, when $a \neq 0$ it possesses a part that evolves as a Brownian motion, and in simple cases, when ν does not vanish, it has a part that consists of jumps. One can show that the discussion presented above extends to finite measures, but in this case the sizes of the jumps may be chosen from a continuum.

Generally, the measure ν determines the sizes of the jumps and the rate at which they can occur. For any Lévy measure with infinite mass, from the definition, we have that the Lévy measure of any such distribution has finite mass on any closed interval that does not contain zero, and charges every neighbourhood of zero with infinite mass. This intuitively suggests that the the jumps that occur with the greatest intensity are small, so that this part of the process is dominated by arbitrarily small jumps. We have already seen that there are at most finitely many jumps greater than any fixed size, and we have that the jumps of any Lévy process with infinite Lévy measure are dense in $[0, \infty)$.

Theorem (Sato, 1999, Theorem 21.3). *If $\nu(\mathbb{R}) = \infty$, then almost surely jumping times are countable and dense in $[0, \infty)$. If $0 < \nu(\mathbb{R}) < \infty$, then almost surely, jumping times are infinitely many, and countable, in increasing order.*

To obtain a somewhat better intuition for the jumps of Lévy processes when the Lévy measure is infinite and for the Lévy-Itô Decomposition, following Kyprianou, we notice that we can express an infinite Lévy measure as sum of finite ones:

$$\int_{0<|x|<1}(1-e^{iut}+iut)\nu(dy)=\sum_{n=0}^{\infty}\left\{\lambda_n\int_{2^{-(n+1)}\le|x|<2^{-n}}(1-e^{iut})F_n(dy)\right.$$
$$\left.+iu\lambda_n\int_{2^{-(n+1)}\le|x|<2^{-n}}xF_n(dy)\right\},$$

where

$$\lambda_n=\nu(\{x:2^{-(n+1)}\le|x|<2^{-n}|\})\quad\text{and}\quad F_n(dy)=\lambda_n^{-1}\nu(dy)|_{x:2^{-(n+1)}<|x|<2^{-n}}.$$

This interpretation leads one to to intuitively regard the part of a Lévy process that is generated by a Lévy measure with infinite mass as a superposition of infinitely many processes with finite Lévy measures; *i.e.* jump processes. The Lévy-Itô Decomposition makes this intuition precise. This is a subtle point. It is not the case that a general Lévy process with neither a drift nor a Brownian motion part is generated by the sum of its jumps (Cont and Tankov, 2003, Remark 3.1). This is because the sum of the small jumps in a Lévy process need not converge.

We will not formally state the general version of Lévy-Itô Decomposition here, as we do not require it, and the mathematics needed to state the theorem will take us far what we need. However, informally, the Lévy-Itô decomposition gives us that a Lévy process $\{X(t):t\ge 0\}$ can be expressed as the the sum of four independent terms:

$$X(t)=bt+B_a(t)+X^1(t)+\lim_{\varepsilon\to 0}\tilde{X}^{\varepsilon}(t),$$

the first is linear drift, the second is Brownian motion with $B_a(t)\sim\mathcal{N}(0,at)$ and the third is a compound Poisson process. The fourth term is called the *compensated sum of jumps*. It is a martingale, and it is centered ($\tilde{X}^{\varepsilon}(t)$ is an integral of the difference between a Poisson random process and its intensity), so it can be thought of an infinite superposition of Poisson processes which can be proved to converge.

In the next section, we state a restricted version of the Lévy-Itô Decomposition.

2.3.3 Subordinator processes

Subordinators are increasing Lévy processes. This immediately implies that they have no Brownian part, and they have bounded variation. Indeed this gives us that the sum of small jumps of subordinator processes converges. Consequently, the following version of the

Lévy-Itô Decomposition holds for subordinators, and we see that subordinators without drift can be regarded as the sum of their jumps.

Theorem: Lévy-Itô Decomposition for Subordinators (Cont and Tankov, 2003, Corollary 3.1) *Let* $\{X(t) : t \geq 0\}$ *be a subordinator. Then*

$$X(t) = bt + \sum_{\tau_j \leq t} dX_{\tau_j},$$

where τ_j *are the jump times of* X *and* dX_{τ_j} *is the size of the jump at time* τ_j

Moreover, since they have no negative jumps, the support of their Lévy measure is contained in $[0,\infty)$. Consequently, the Lévy-Khintchine formula for the characteristic function for a subordinator, $W(t) : t \geq 0$, simplifies, Bertoin (1996):

$$\mathbb{E}\Big(e^{iuW(t)}\Big) = \exp\Big(ibu + t\Big[\int_{(0,\infty)} (e^{iuw} - 1)\nu(dw)\Big]\Big),$$

where the Lévy measure ν satisfies,

$$\int_{(0,\infty)} (1 \wedge x)\nu(dx) < \infty.$$

This result is an application of the Lévy-Khintchine Formula and the Lévy-Itô decomposition, and it uses the fact that subordinators have finite variation.

2.3.4 Generalized Inverse Gaussian Lévy processes.

Generalized inverse Gaussian Lévy processes are Lévy processes whose values at time one are distributed accored to a generalized inverse Gaussian distribution. They are pure jump Lévy process whose jump sizes are always positive, and so they are subordinators.

The Lévy density for a GIG process is given by Eberlein and Hammerstein (2004):

$$Q_{GIG}(x) = \frac{e^{-x\gamma^2/2}}{x}\left[\int_{(0,\infty)} \frac{e^{-xy}}{\pi^2 y|H_{|\lambda|}(\delta\sqrt{2y})|^2}\,dy + \max(0,\lambda)\right], \quad \text{for } x > 0.$$

The function $H_\nu(z)$ is the Hankel function of the first kind: $H_\nu(z) = J_\nu(z) + iY_\nu(z)$, where J, Y are Bessel functions of the first and second kinds, respectively. We will only need to consider these functions on the positive reals.

From the formula for the Lévy density, one sees that for every $t > 0$, the interval $(0,t)$ has infinite Lévy measure. We have seen that this implies that in every interval of positive length, a GIG process almost surely has infinitely many jumps.

To avoid confusion, it is worth pointing out that while GIG distributions are infinitely divisible, and the restriction of a GIG process to each subinterval of its domain is a GIG process as well, it is not the case that the distributions of the time t values of a GIG process have the same parameters as its time one distribution.

In this work, our subordinators will always be generalized inverse Gaussian Lévy processes.

2.3.5 Generalized hyperbolic processes

The Lévy-Khintchine representation of the characteristic function of a GH distribution is given by Eberlein (2001)

$$\phi_{GH}(x) = \exp\left[iuE(GH) + \int_{-\infty}^{\infty}\Big(e^{iux} - 1 - iux\Big)g(x)\,dx\right], \quad \text{where } \zeta = \delta\sqrt{\alpha^2-\beta^2},$$

and

$$E(GH) = \mu + \frac{\beta\delta^2}{\zeta}\frac{K_{\lambda+1}(\zeta)}{K_\lambda(\zeta)},$$

is the first moment of the GH distribution, and g is the density of the Lévy measure, which we include for completeness,

$$g(x) = \frac{e^{\beta x}}{|x|}\left(\int \frac{\exp\Big(-\sqrt{2y+\alpha^2}|x|}{\pi^2 y\big(J_\lambda^2(\delta\sqrt{2y}) + Y_\lambda^2(\delta\sqrt{2y})\big)}\,dy + \lambda e^{-\alpha|x|}\right), \text{ if } \lambda >= 0$$

and

$$g(x) = \frac{e^{\beta x}}{|x|}\left(\int \frac{\exp\Big(-\sqrt{2y+\alpha^2}|x|}{\pi^2 y\big(J_{-\lambda}^2(\delta\sqrt{2y}) + Y_{-\lambda}^2(\delta\sqrt{2y})\big)}\,dy\right), \text{ if } \lambda < 0$$

Since a GH distribution $X = \{X(t) : t \geq 0\}$ is a normal mean-variance mixture with mixing distribution a GIG $W = \{W(t) : t \geq 0\}$, when $\mu = \beta = 0$, we may express

$$dX_{\tau_i} = \alpha_i\sqrt{dW_{\tau_i}}, \quad \text{where } \alpha_i \sim \mathcal{N}(0,1),$$

where the jump times of X and W coincide and are $\{\tau_i\}_{i=1}^{\infty} \subset [0,\infty)$, for each τ_i, dX_{τ_i} is the size of the jump at time τ_i, and dW_{τ_i} is the size of the jump of W.

2.4 The Gaussian Process Convolution Model

We are now able to completely describe the Lévy driven GPCM which will be the topic of Chapter 4.

We will assume that $X = \{X(\tau) : \tau \geq 0\}$ a GH process that is obtained as a normal variance-mixture of W, of a GIG subordinator $W = \{W(\tau) : \tau \geq 0\}$. For simplicity, we will assume that the jump X_{τ_i} of at time τ_i can be expressed as

$$X_{\tau_i} = \alpha_i \sqrt{W_{\tau_i}} \quad \text{where} \quad \alpha_j \sim \mathcal{N}(0,1).$$

We let $h \sim GP(0, \mathcal{K}_{DEQ}(t_1,t_2))$ be the filter for the model.

Now the formula for the Lévy driven GPCM is

$$f(t) = \int_{-\infty}^{\infty} h(t-\tau)dX(\tau) = \sum_j h(t-\tau_j)dX_{\tau_j} = \sum_j h(t-\tau_j)\alpha_j dW_{\tau_j}.$$

2.5 Markov Chain Monte Carlo (MCMC) Methods

Markov chain Monte Carlo methods are an essential tool for sampling from unknown, high-dimensional distributions. In this situation, thoroughly exploring the domain by hand, using for example fine grid of points, is computationally infeasible.

In this section, we will describe the Metropolis-Hastings Algorithm, and an important special case, Gibbs sampling, which are the tools that we will use for inference for Lévy driven GPCMs. We refer the reader to Murphy (2023) for further background on these methods.

2.5.1 Metropolis-Hastings Algorithm

The Metropolis-Hastings Algorithm was first published in Metropolis et al. (1953), which dealt with the case of symmetric proposal distributions and was extended to the general case in Hastings (1970).

It is common to encounter probability distributions that one only knows up to a multiplicative factor. When this occurs, the Metropolis-Hastings Algorithm, provides us with a method for generating samples which are (approximately) from the unknown distribution.

Suppose that $Cq(x) = p(x)$, where p is an unknown, but q is known. To generate samples from p one proceeds as follows: Starting at a random point in parameter space, one explores the space by selecting a potential next point x' from a proposal distribution $Q(x'|x)$, and then deciding whether to accept, *i.e.* move to, that point, or not, and remain in the current position.

One accepts the proposal with probability

$$a = \min\left(1, \frac{q(x')Q(x|x')}{q(x)Q(x'|x)}\right).$$

When the proposal distribution is symmetric, so that $Q(x'|x) = Q(x|x')$, this has a simple interpretation: we move to the new location whenever $p(x') \geq p(x)$, and with probability $p(x')/p(x) = q(x')/q(x)$, when $p(x') < p(x)$. When the proposal is asymmetric, we must compensate for the fact that the proposal distribution may favour certain states.

This selection process generates a Markov chain $x_1, x_2, x_3, \ldots$ whose stationary distribution is p. Provided that the transition matrix for the Markov chain is ergodic and irreducible, for any starting point x_1, the samples from the Markov chain eventually approximate samples from p, and one says that the chain converges to the stationary distribution. The time that it takes the chain to converge to the stationary distribution is known as the *burn in time*. In practice, the burn in time is difficult to estimate; however examining the errors of the samples can give an indication as to whether the chain has converged. To check whether the chain has converged, one should also consider the Markov chain generated from different starting points, and check whether the limiting distributions are the same. Taking N to be a suitable burn in time, the samples $x_N, x_{N+1}, \ldots$, can be regarded as correlated samples from p. To make estimates more robust it is sometimes helpful to thin this collection, taking every k^{th} sample for examine to obtain a collection that is less correlated.

2.5.2 Gibbs sampling

Gibbs sampling was introduced in Geman and Geman (1984). It is analogous to coordinate ascent. Gibbs sampling useful when one is unable to sample from a joint distribution $p(x_1, x_2, \ldots, x_n)$, but is able to sample from each of the conditional distributions, $p(x_i|x_1, \ldots, x_{i-1}, x_{i+1}, \ldots, x_n)$, for $i = 1, 2, \ldots, n$. Sampling successively in this way generates approximate samples from the joint distribution. It turns out that Gibbs sampling is a special case of Metropolis-Hastings, and regarded as a Metropolis-Hastings algorithm the acceptance probability 1. As with the Metropolis-Hastings algorithm, one should consider the samples after some burn in time, and to reduce correlation between the samples in may be helpful to thin them.

Chapter 3

Simulation of Lévy processes

Series representations of Lévy processes provide a powerful and principled tool for obtaining approximations of these stochastic processes. For our computations, approximations are inevitably required as the Lévy processes we consider almost surely have infinitely many jumps in every interval. However, the only approximation to Lévy processes that we make is a choice of the truncation of the series.

3.1 Shot-noise representations of Lévy processes

Let $W = \{W(t) : t \geq 0\}$ denote a subordinator Lévy process with no drift or Brownian motion part. Then it follows from the Lévy-Itô decomposition and the Levy-Khintchine Theorem that we can express the characteristic function of W as

$$\mathbb{E}(e^{iuW(t)}) = \exp\left[t \int_{(0,\infty)} (e^{iuw} - 1) Q(dw)\right],$$

where Q is a Lévy measure on $\mathbb{R} \setminus 0$ satisfying $\int_{0,\infty}(1 \wedge x)Q(dx) < \infty$, and $1 \wedge x$ denotes the minimum of 1 and x. By the Lévy-Ito integral representation, almost surely, for each $t \geq 0$, we may express

$$W(t) = \int_{(0,\infty)} wN([0,t], dw),$$

where N is the point process of jumps of W; that is,

$$N = \sum_{i=1}^{\infty} \delta_{V_i, W_i},$$

where $V_i \in [0,T]$ are i.i.d. uniform random variables giving the arrival times of the jumps, W_i are the sizes of the jumps, and δ_{V_i,W_i} is a Dirac measure centered on (V_i,W_i). Thus we obtain

$$W(t) = \sum_{i=1}^{\infty} W_i \chi_{V_i \leq t}.$$

Theorem (Rosiński (2001)). *The series $\sum_{i=1}^{\infty} W_i \chi_{V_i \leq t}$ converges almost surely.*

3.1.1 Simulating Lévy processes

In practice, since there are almost surely infinitely many jumps in any time interval, it is not possible to simulate a subordinator Lévy processes directly from N. Instead one uses the following approach of Rosiński (2001); Wolpert and Ickstadt (1998) and Ferguson and Klass (1972).

Recall that the sum of n independent exponential random variables with unit rate is distributed according to Gamma$(n,1)$. Hence it is straightforward to simulate the epochs of a unit rate Poisson process $\{\Gamma_i\}$: generate standard exponential random variables and calculate their cumulative sums.

We let $Q^+(x) = Q([x,\infty])$, denote the *upper tail probability* of the Lévy measure, and define the *inverse tail probability*, $h^{-1}(\gamma) = (Q^+)^{-1}(\gamma)$. Then by Rosinski's Theorem, the the point process $\sum_{i=1}^{\infty} \delta_{V_i,h(\Gamma_i)}$ converges almost surely to N. Where $\sum_{i=1}^{\infty} \delta_{V_i,h(\Gamma_i)}$ is a Poisson point process on $[0,T] \times [0,\infty)$. s before, the V_i are uniformly distributed jump times and $h(\Gamma_i)$ is the size of the jump at time V_i. Even more, the sizes of the jumps decreases with i, so that truncating this series corresponds to discarding the smallest jumps.

Unfortunately, it is not possible to compute h explicitly, so a thinning (or rejection sampling) approach is adopted, Lewis and Shedler (1979), Rosiński (2001): The strategy is to find bounding process N_0, which we know how to sample from, with Lévy measure Q_0 satisfying $dQ_0(x)/dQ(x) \geq 1$ for all $x > 0$. Then samples from N_0 are thinned with probability $dQ(x)/dQ_0(x)$ to obtain samples from the desired distribution N.

The bounding processes that are used in these simulations are the tempered stable processes and the Gamma process.

Tempered stable point processes

The Lévy density for the tempered stable process is given by

$$Q(x) = Cx^{-1-\alpha}e^{-\beta x},$$

observe that it factors into a positive α-stable process with Lévy density $Q_0(x) = Cx^{-1-\alpha}$ and tempering function $e^{-\beta x}$. Computing the integral, the tail mass of Q_0 is given by $Q_0^+(x) = \frac{C}{\alpha}x^{-\alpha}$. So to generate a tempered stable process one proceeds as follows, Godsill and Kındap (2021):

1. Set $N = \emptyset$
2. Generate the epochs of a unit rate Poisson process $\{\Gamma_i : i \in \mathbb{N}\}$.
3. For each $i \in \mathbb{N}$:
   (a) Compute $x_i = \left(\frac{\alpha\Gamma_i}{C}\right)^{-1/\alpha}$.
   (b) With probability $e^{-\beta x_i}$, accept x_i and set $N = N \cup \{x_i\}$.
4. For each $x_i \in N$, generate a jump time v_i uniformly in $[0,T]$.
5. Obtain a realization of the tempered stable Lévy process $w(s) = \sum_{i=1}^{\infty} x_i \chi_{v_i \leq s}$.

Gamma processes

The Lévy density of a Gamma process is given by

$$Q(x) = Cx^{-1}e^{-\beta x}.$$

As for tempered stable processes, we simulate Gamma processes using thinning. We take a dominating point process with Lévy measure

$$Q_0 = \frac{C}{x}(1+\beta x)^{-1},$$

and we can compute its tail probability $Q_0^+(x) = C\log(\beta^{-1}x^{-1}+1)$, and $h(\gamma) = \frac{1}{\beta(\exp(\gamma/C)-1)}$. Points are thinned with probabilty $Q(x)/Q_0(x) = (1+\beta x)e^{-\beta x}$.

3.1.2 Simulation of generalized inverse Gaussian processes

In this section we briefly present the main results of Godsill and Kındap (2021) which provide a thinning approach for obtaining realizations of generalized inverse Gaussian processes.

Recall that the Lévy density for a GIG process is given by Eberlein and Hammerstein (2004)

$$Q_{GIG}(x) = \frac{e^{-x\gamma^2/2}}{x}\left[\int_{(0,\infty)} \frac{e^{-xy}}{\pi^2 y|H_{|\lambda|}(\delta\sqrt{2y})|^2}\,dy + \max(0,\lambda)\right], \quad \text{for } x > 0.$$

Notice that the GIG Lévy density is a superposition of two densities: The term $\max(0,\lambda)\frac{e^{-x\gamma^2/2}}{x}$ is the Lévy density of a Gamma process, which we have already seen how to sample from, so we will focus on the term

$$Q(x) = \frac{e^{-x\gamma^2/2}}{x}\int_{(0,\infty)} \frac{e^{xy}}{\pi^2 y|H_{|\lambda|}(\delta(\sqrt{2y}))|}\,dy,$$

which we will find convenient to express as

$$\frac{2e^{-x\gamma^2/2}}{\pi^2 x}\int_{(0,\infty)} \frac{\exp\left(\frac{-z^2x}{2\delta^2}\right)}{x|H_{|\lambda}(z)|^2}\,dz.$$

We let

$$Q_{GIG}(z,x) = \frac{2}{\pi^2 x}\frac{\exp\left(-\frac{x\gamma^2}{2}-\frac{z^2x}{2\delta^2}\right)}{z|H_{|\lambda|}(z)|^2},$$

denote a bivariate intensity function associated with a point process on $(0,\infty)\times(0,\infty)$. It is important to observe that $Q(x) = \int_0^\infty Q(x,z)\,dz$.

We will give the details of the method of Godsill and Kındap (2021) for sampling from a GIG process when $\lambda \geq 0.5$. Theorem 1 of that paper gives us that

$$Q_{GIG}(x,z) \leq \frac{e^{-x\gamma^2/2}}{\pi x}e^{-\frac{z^2x}{2\delta^2}},$$

which may be expressed as (Godsill and Kındap, 2021, Corollary 1)

$$Q_{GIG}(x,z) \leq \frac{\delta\Gamma(1/2)e^{-x\gamma^2/2}}{\sqrt{2}\pi x^{3/2}}\sqrt{Ga}\left(z\middle|\frac{1}{2},\frac{x}{2\delta^2}\right) =: Q^0_{GIG}(x,z),$$

where We let $\Gamma(x)$ denote the Γ-function:

$$\Gamma(x) = \int_0^\infty t^{x-1}e^{-t}\,dt,$$

and the density $\sqrt{Ga}$ denotes the square-root Gamma density. Its density function is given by

$$\sqrt{Ga}(z|\alpha,\beta) = \frac{2\beta^\alpha}{\Gamma(\alpha)}x^{2\alpha-1}e^{-\beta x^2},$$

and it is the the density of a random variable $X^{1/2}$ when $X \sim Ga(x|\alpha,\beta)$.

Notice that the first factor in the upper bound:

$$\frac{\delta\Gamma(1/2)e^{-x\gamma^2/2}}{\sqrt{2}\pi x^{3/2}}$$

is the intensity function of a tempered stable process, which is a process that we know how to sample from. We also have that

$$\frac{Q_{GIG}(x,z)}{Q^0_{GIG}(x,z)} = \frac{2}{\pi z|H_{|\lambda|}(z)|^2}.$$

Thus we have the following algorithm for sampling from the GIG distribution when $\lambda \geq 1/2$:

1. Generate many samples from the tempered stable process with $C = \delta\Gamma(1/2)/\sqrt{2}\pi, \alpha = 1/2$ and $\beta = \gamma^2/2$.

2. For each sampled point x, sample

$$z \sim \sqrt{Ga}\Big(z\Big|\frac{1}{2}, \frac{x}{2\delta^2}\Big).$$

   This gives us a sample (x,z) from the dominating process $Q^0_{GIG}(x,z)$.

3. For each pair (x,z), accept with probability

$$\frac{2}{\pi z|H_{|\lambda|}(z)|^2}.$$

When $\lambda < 0.5$, we no longer have a dominating process Q^0_{GIG} which factors so nicely, and more sophisticated (piece-wise) bounds used are used to construct a dominating process for these GIG processes. In Kındap and Godsill (2022a), these more sophisticated bounds were extended to the $\lambda \geq 0.5$ case. Moreover, Kındap and Godsill (2022a) develops an adaptive truncation procedure to select the number of jumps to be used in the approximation.

3.2 Experiments

In this section we will demonstrate the sampling algorithms for sampling paths of generalized hyperbolic Lévy processes. I partially implemented the sampling algorithms of Kındap and Godsill (2022a) using the available code at https://github.com/yamankindap/GiG as a base.

As in Godsill and Kındap (2021); Kındap and Godsill (2022a) we use three tools for evaluating how well we are generating samples of GH processes:

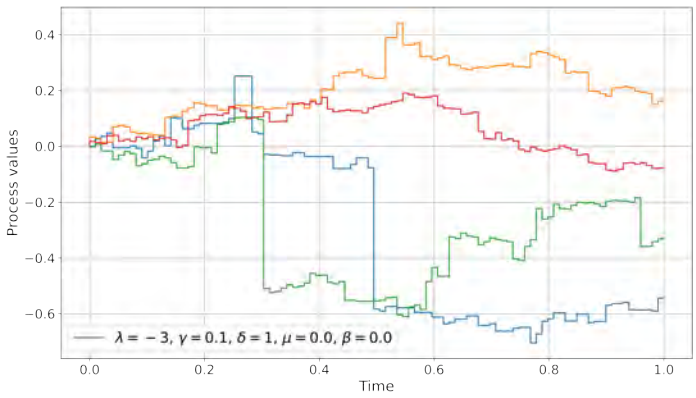


Fig. 3.1 Plots of the generalized hyperbolic process with $\lambda = -3, \delta = 1$ and $\gamma = 0.1$.

- We visually compare distributions of the time one values of the sample paths with the expected densities.
- We analyze the Q-Q-plots of time one values of the sample paths with points sampled from the expected GH distribution. Q-Q-plots are quantile-quantile plots; they are obtained by plotting the quantiles of one distribution against the quantiles of a second. A point (x,y) on the curve indicates that x,y are the q-th quantiles both the first and second distribution, respectively. When two distributions are similar, the plot will be close to the graph of the identity.
- We evaluate the KS-statistic of values from the sample paths and samples from the expected distribution. The KS-statistic is named for Kolmogorov-Smirnov. It is an estimate of the distance between two empirical distributions, which measures the probability that the two sets of samples were drawn from the same distribution. The KS-test reports a statistic together with a p-values, the statistic is the supremum of the distances between the CDFs of the distributions. The null hypothesis for the KS-test is that the samples are drawn from the expected distribution, so that large p-values indicate that the null hypothesis cannot be rejected, and it may be that the samples are from the same distribution. In our experiments we considers samples of size 10000. At this level it is sufficient to have a p-value of at least 0.1.

Let us fix parameters $\delta = 1$, $\gamma = 0.1$ and evaluate the perfomance of the sampling algorithm as λ varies. We can see this very nicely in the plots of the sample paths. For example Figure 3.1, 3.2 and 3.3 the value of λ increasing as we see in the pictures that the effect of small jumps diminishes as λ increases and large jumps dominate.

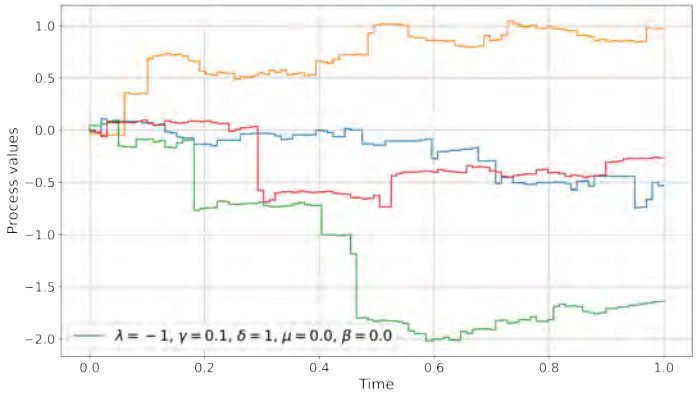


Fig. 3.2 Plot of the generalized hyperbolic process with $\lambda = -1, \delta = 1$ and $\gamma = 0.1$.

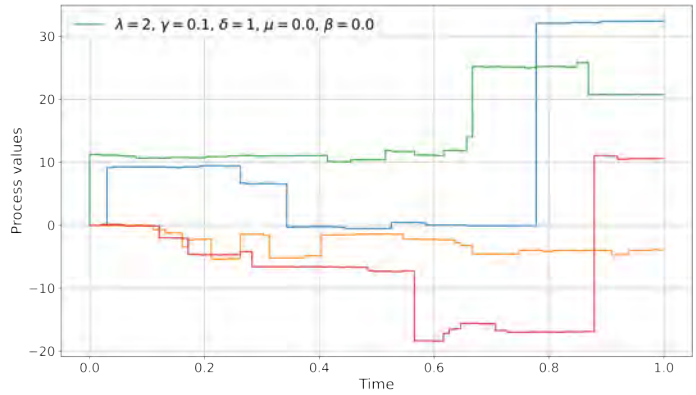


Fig. 3.3 Process plots for a GH process with parameters $\lambda = 2, \delta = 1$ and $\gamma = 0.1$

λ	# Samples	KS-statistic	p-value
3	10000	0.01	0.63
2	10000	0.01	0.21
1	10000	0.007	0.8
0.3	10000	0.006	0.91
-0.3	10000	0.11	0.26
-1	10000	0.012	0.14
-2	10000	0.009	0.45
-3	10000	0.01	0.29

Table 3.1 Using the KS-statistic to compare the time one values of Lévy process simulations with samples from GH distributions for different values of λ.

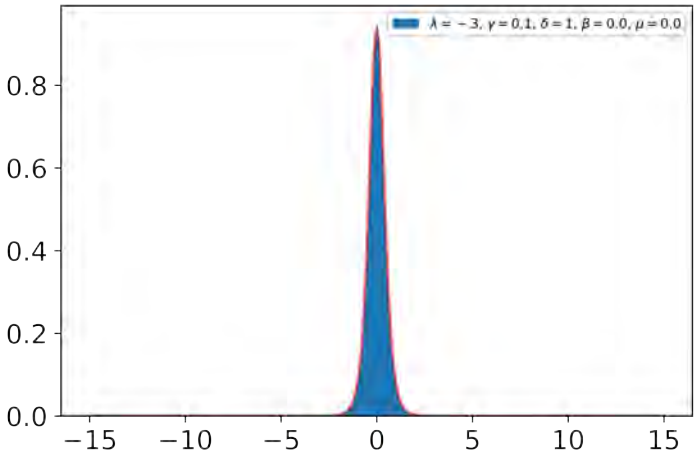


Fig. 3.4 Histogram for the time one values of a GH process with parameters $\lambda = -3, \delta = 1$ and $\gamma = 0.1$ against samples from the corresponding GH-density.

Now, let us consider the question of how well the time one values of processes approximate the expected distribution. In Table 3.1, we see that that in each example, the KS-statistic is fairly small, and the p-value is always great enough that we cannot reject the null hypothesis that the distribution of the time one samples is drawn from the expected GH distribution. The worst performing parameter value is $\lambda = -0.3$. It is perhaps surprising that there is no obvious patten behind which parameter values lead to poor or good performance.

If we examine the Q-Q plots and the histograms we see that they the distributions match quite well, see Figure 3.4 and 3.5. For the Q-Q plots, we plot the 0.005 quantile to the 0.995 quantile. For smaller or larger values we do have enough data to obtain an accurate picture; however, we did not notice any systematic bias in our plots that contained quantiles out to the boundaries.

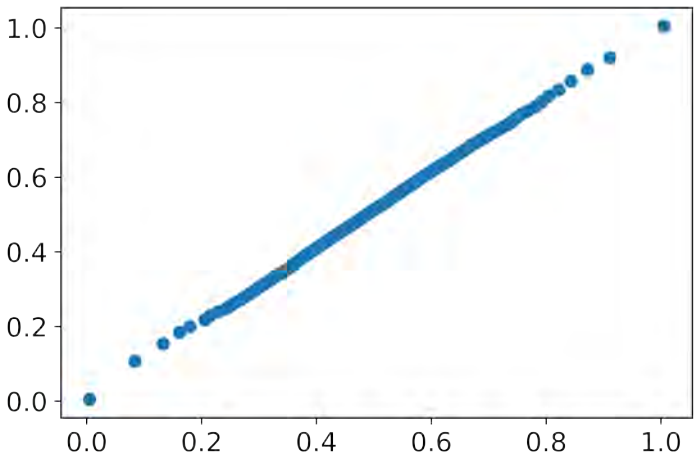


Fig. 3.5 Q-Q plot for the time one values of a GH process with parameters $\lambda = -3, \delta = 1$ and $\gamma = 0.1$ against samples from the corresponding GH-density.

We refer the reader to Godsill and Kındap (2021); Kındap and Godsill (2022a) for further information about these processes and demonstrations that seem to producing values from the expected distributions.

Chapter 4

Inference for GPCMs driven by Generalized Hyperbolic Lévy Processes

In this chapter, we develop tools for inference for Gaussian Process Convolution Models. Let us recall that our model assumes that our data points $\{(t_i, y_i)\}_{i=1}^{N_d}$ are generated by noisy observations from a process

$$f(t) = \int_{\mathbb{R}} h(t-\tau) dX(\tau) = \sum_j h(t-\tau_j) dX_{\tau_j},$$

where $h \sim GP(0, \mathscr{K}_{DEQ}(t_1, t_2))$ and $X = \{X(\tau) : \tau \geq 0\}$ is a GH Lévy process with jumps dX_{τ_j} at times τ_j, and the sum is taken over all jumps of X. We assume that

$$y_i = f(t_i) + \varepsilon_i \quad \text{where } \varepsilon_i \sim \mathscr{N}(0, \sigma_{err}^2).$$

Since X is a GH Lévy process, there exist constants $\mu, \beta \in \mathbb{R}$, and a subordinator GIG Lévy process $W = \{W(\tau) : \tau \geq 0\}$, so that the jumps of X can be expressed as

$$dX_{\tau_i} = \mu + \beta W_{\tau_i} + \alpha_i \sqrt{W_{\tau_i}} \quad \text{where } \alpha_i \sim \mathscr{N}(0,1). \tag{4.1}$$

We will perform inference using Gibbs sampling. Our objective is to generate samples of the joint distribution $p(W, \alpha, h|\boldsymbol{y})$, by sampling from the marginal distributions $p(h|W, \alpha, \boldsymbol{y})$, $p(\alpha|W, h, \boldsymbol{y})$ and $p(W|h, \boldsymbol{y})$, where we let α denote the set of *jump coefficients* from equation (4.1). Observe that in the last step we perform *collapsed Gibbs sampling* (see, for example, Murphy (2023)) and integrate out α. The flexibility of our model makes it possible for the choices of α and h to compensate for deficiencies in the sampled subordinator, which leads to poor exploration of the space of subordinators. To reduce the effect of this problem,

when we sample the subordinator distribution, instead of sampling from $p(W|h,\alpha,\boldsymbol{y})$, we sample from $p(W|h,\boldsymbol{y}) = \int p(W|h,\boldsymbol{\alpha},\boldsymbol{y})p(\alpha)\,d\alpha$. We will see that the result is a Gibbs sampler which seems to explore the space of subordinator processes well.

Observe that the distributions $p(h|W,\alpha,\boldsymbol{y})$ and $p(\alpha|W,h,\boldsymbol{y})$ are Gaussian and straightforward to sample from; however, sampling entire realizations of W from $p(W|h,\boldsymbol{y})$ produces a Gibbs sampler that is slow to converge and we will make use of a Metropolis-Hastings within Gibbs algorithm to sample the jumps in small intervals from a partition of the domain of W. This algorithm was used before in a similar context in Kındap and Godsill (2022b)).

We will often consider the density $p(\boldsymbol{y}|W,\alpha,h)$, which we will sometimes denote as $p(\boldsymbol{y}|f)$, where $f(t) = \sum_j h(t-\tau_j)\alpha_j\sqrt{W_{tau_j}}$, since together W,α,h determine the convolution f.

4.1 Preliminaries

4.1.1 Parametrizing the filter

In Bruinsma et al. (2022); Tobar et al. (2015), the pseudo-points approximation of Titsias (2009) is used to parametrize the filters of GPCMs. We experimented with using them to parametrize the filters of our GPCMs, and with parametrizing the filters using their values at a specifed set of points, *inducing times*, and inferring a filter from the posterior Gaussian process using noiseless Gaussian process regression. We found that the second method worked well, so that is how we decided to parametrize Gaussian processes. However there are likely performance benefits to be gained from using the pseudo-points approximation.

4.1.2 Integrating out the jump coeffcients α

The calculations in this section are standard, see for example Cemgil (2001). The techniques used here will be used to integrate out α to obtain $p(W|h,\boldsymbol{y})$, to find the Gaussian form of $p(h|W,\alpha,\boldsymbol{y})$ and to optimize hyper-parameters of the Gaussian process, from which h is drawn.

Using the truncated shot-noise representation of the subordinator process, we approximate the convolution by a finite sum $f(t) \approx \sum_{j=1}^{M} \alpha_j h(t-\tau_j)\sqrt{W_{\tau_j}}$. Since for a data point $y_i = f(t_i) + \varepsilon_i$, where $\varepsilon_i \sim \mathcal{N}(0,\sigma_{err}^2)$, we have $y_i \sim \mathcal{N}(f(t_i),\sigma_{err}^2)$, so that

$$p(y_i|f) \approx \frac{1}{\sqrt{2\pi}\sigma_{err}} \exp\left[-\frac{1}{2\sigma_{err}^2}\Big(y_i - \sum_{j=1}^{M} \alpha_j h(t_i-\tau_j)\sqrt{W_{\tau_j}}\Big)^2\right].$$

Let $\omega_j = h(t_i - \tau_j)\sqrt{W_{\tau_j}}$. Setting $\alpha = (\alpha_1, \ldots, \alpha_M)$, $\mathbf{m} = (\frac{y_i}{\omega_1 M}, \ldots, \frac{y_i}{\omega_M M})$, and Σ^{-1} to be the $M \times M$ matrix with (j,k)-th entry $\omega_j \omega_k / \sigma_{err}^2$, we have that

$$(\alpha - \mathbf{m})^T \Sigma^{-1} (\alpha - \mathbf{m}) = \frac{1}{\sigma_{err}^2} \Big(y_i - \sum_{j=1}^{M} \alpha_j h(t_i - \tau_j) \sqrt{W_{\tau_j}} \Big)^2.$$

We consider the vector $\mathbf{m}$ purely formally, since it may happen that some ω_j is zero; however, we can always arrange our computations so that there are cancellations and dividing by zero never happens.

Now, since each $\alpha_j \sim \mathscr{N}(0,1)$, we can approximate

$$p(y_i|f)p(\alpha) \approx \frac{1}{(2\pi)^{(M+1)/2} \sigma_{err}} \exp \left[-\frac{1}{2} \Big((\alpha - \mathbf{m})^T \Sigma^{-1} (\alpha - \mathbf{m}) \Big) - \frac{1}{2} \alpha^T \alpha \right].$$

In what follows we will need to have a formula for $\det(\mathbb{I}_M + \Sigma^{-1})$. Notice that $\Sigma^{-1} = \omega_0 \omega_0^T$, where $\omega_0 = \omega / \sigma_{err}$. Now,

$$(\mathbb{I}_M + \Sigma^{-1})\omega_0 = \omega_0 + \omega_0 \omega_0^T \omega_0 = (1 + \omega_0^T \omega_0)\omega_0,$$

so we have that ω_0 is an eigenvector of $\mathbb{I}_M + \Sigma^{-1}$ with eigenvalue $1 + \omega_0^T \omega_0$. Since $\mathbb{I}_M + \omega_0 \omega_0^T$ is real-symmetric, it has a basis of mutually orthogonal eigenvectors. If u is another eigenvector, it is orthogonal to ω_0, so that $(\omega_0 \omega_0^T + \mathbb{I}_M)u = u$, and hence u has eigenvalue 1. Since the determinant of a matrix is the product of its eigenvalues, we have that

$$\det(\Sigma^{-1} + \mathbb{I}_M) = 1 + \omega_0^T \omega_0 = 1 + \omega^T \omega / \sigma_{err}^2.$$

Moreover, since $\omega_0^T \omega_0 = \|\omega_0\|^2 \geq 0$, the determinant of $\mathbb{I}_M + \omega \omega^T$ does not vanish, and so $\mathbb{I}_M + \Sigma^{-1}$ is invertible.

Now, taking

$$\widehat{\mathbf{m}} = (\mathbb{I}_M + \Sigma^{-1})^{-1} \Sigma^{-1} \mathbf{m}$$

we have that

$$(\alpha - \widehat{\mathbf{m}})^T (\mathbb{I}_M + \Sigma^{-1})(\alpha - \widehat{\mathbf{m}}) + \mathbf{m}^T \Sigma^{-1} \mathbf{m} - \mathbf{m}^T \Sigma^{-1} (\mathbb{I}_M + \Sigma^{-1})^{-1} \Sigma^{-1} \mathbf{m}$$

$$= (\alpha - \mathbf{m})^T \Sigma^{-1} (\alpha - \mathbf{m}) \Big) - \alpha^T \alpha.$$

Note that

$$\mathbf{m}^T \Sigma^{-1} \mathbf{m} = y_i^2 / \sigma_{err}^2.$$

Setting

$$C = y_i^2/\sigma_{err}^2 - \mathbf{m}^T\Sigma^{-1}(\mathbb{I}_M + \Sigma^{-1})^{-1}\Sigma^{-1}\mathbf{m},$$

we can express

$$\frac{1}{2\pi\sigma_{err}}\exp\left[-\frac{1}{2}\Big((\alpha-\mathbf{m})^T\Sigma^{-1}(\alpha-\mathbf{m})\Big) - \frac{1}{2}\alpha^T\alpha\right]$$

$$= \frac{1}{(2\pi)^{(M+1)/2}\sigma_{err}}\exp\left[-\frac{1}{2}\Big((\alpha-\widehat{\mathbf{m}})^T(\Sigma^{-1}+\mathbb{I}_M)(\alpha-\widehat{\mathbf{m}}) + C\Big)\right].$$

Since it is a probability density, we have that

$$\int_{\mathbb{R}^M} \frac{1}{\sqrt{\det(2\pi(\Sigma^{-1}+\mathbb{I}_M)^{-1})}}\exp\left[-\frac{1}{2}\Big((\alpha-\widehat{\mathbf{m}})^T(\Sigma^{-1}+\mathbb{I}_M)(\alpha-\widehat{\mathbf{m}})\Big)\right] d\alpha = 1.$$

Thus we can estimate

$$\int p(y_i|f)p(\alpha)\,d\alpha \approx \frac{\sqrt{\det(2\pi(\Sigma^{-1}+\mathbb{I}_M)^{-1})}}{(2\pi)^{(M+1)/2}\sigma_{err}}\exp\left[-\frac{1}{2}\mathbf{C}\right].$$

Thus we see that we can integrate out α analytically.

For calculation purposes the following observations are very useful.

We have that

$$\sqrt{\det(2\pi(\Sigma^{-1}+\mathbb{I}_M)^{-1})} = \frac{(2\pi)^{M/2}}{\sqrt{\det(\Sigma^{-1}+\mathbb{I}_M)}}.$$

To compute C, we use the following two identities. First, multiplying, one finds that

$$\Sigma^{-1}\omega = \frac{y_i\omega}{\sigma_{err}^2}.$$

Second, to compute $(\mathbb{I}_M + \Sigma^{-1})^{-1}$,

$$(\mathbb{I}_M + \Sigma^{-1})^{-1} = (\mathbb{I}_M + \omega_0\omega_0^T)^{-1} = \mathbb{I}_M + \frac{1}{1+\omega_0^T\omega_0}\omega_0\omega_0^T = \mathbb{I}_m + \frac{1}{\omega_0^T\omega_0}\Sigma^{-1},$$

where the second equality follows from the Sherman-Morrison Formula.

Hence, we have that

$$\int p(y_i|f)p(\alpha)\,d\alpha = \frac{1}{\sqrt{2\pi}\sigma_{err}}\frac{1}{\sqrt{1+\omega^T\omega/\sigma_{err}^2}}.$$

$$\exp\left[-\frac{1}{2}\left(y\sigma_{err}^2 - \frac{y_i^2}{\sigma_{err}^4}\omega^T\left(\mathbb{I}_M - \frac{1}{1+\omega^T\omega/\sigma_{err}^2}\Sigma^{-1}\right)\omega\right)\right]$$

Algorithm for computing $\log p(\mathbf{y}|W,h)$

Assume that W is approximated by a Lévy process with M jumps.

1. For each data point (t_i, y_i) do the following:

   a. Compute the vector $\omega_i = (h(t_i-\tau_1)\sqrt{W_{\tau_1}}, \ldots, h(t_i-\tau_M)\sqrt{W_{\tau_M}})$

   b. Let $\Sigma_i = (1/\sigma_{err}^2)\omega_i\omega_i^T$ and $\hat{\Sigma}_i = \mathbb{I}_M - \frac{1}{1+\omega_i^T\omega_i/\sigma_{err}^2}\Sigma_i$.

   c. Compute

   $$A_i = \frac{y_i^2}{\sigma_{err}^4}\omega_i^T\hat{\Sigma}_i\omega_i$$

   and set

   $$C_i = \frac{y_i^2}{\sigma_{err}^2} - A_i$$

   d. Let

   $$p_i = -\frac{1}{2}\log(2\pi\sigma_{err}^2) - \frac{1}{2}\log(1+\omega^T\omega/\sigma_{err}^2) - \frac{1}{2}C_i.$$

2. Compute $\log p(\boldsymbol{y}|W,h) = \sum_i p_i$.

It is useful to observe that step 1 of this algorithm can be computed in parallel for all the data points at once.

4.2 Gibbs sampling

In this section we describe each the steps in our Gibbs sampling procedure.

4.2.1 Sampling α given h and W

To sample from $p(\alpha|h,W,\mathbf{y})$. We observe that

$$p(\alpha|h,W,\mathbf{y}) = \frac{p(y|W,h,\alpha)p(\alpha)}{p(y)} \propto p(\boldsymbol{y}|W,h,\alpha)p(\alpha),$$

and, since it is a product is a Gaussian pdfs, we have from Section 4.1.2 that

$$p(\alpha|\boldsymbol{y},W,h) \sim \mathcal{N}(\alpha|\widehat{\boldsymbol{m}},(\mathbb{I}+\Sigma^{-1})^{-1}).$$

Let us give the algorithm for computing this distribution.

Algorithm for computing $\mathcal{N}(\widehat{\mathbf{m}}, (\mathbb{I}+\Sigma)^{-1})$

Suppose that the subordinator W is approximated by one with M jumps.

1. Initialize the prior of α: $K_\alpha = \mathbb{I}_M$ and $\widehat{\boldsymbol{m}} = 0 \in \mathbb{R}^M$.

For each data point (t_i, y_i), do steps 2 and 3:

2. Let $\omega_i = (h(t_i - \tau_1)\sqrt{W_{\tau_1}}, \ldots, h(t_i - \tau_M)\sqrt{W_{\tau_M}})$

3. Let $\hat{\omega} = \omega_i / \sigma_{err}$ Update

$$K_\alpha \leftarrow K_\alpha - \frac{1}{1 + \hat{\omega}_i^T K_\alpha \hat{\omega}_i} K_\alpha \hat{\omega}\hat{\omega}^T K_\alpha$$

$$\widehat{\boldsymbol{m}} \leftarrow \widehat{\boldsymbol{m}} + \frac{y_i}{\sigma_{err}^2}\omega.$$

4. $p(\alpha|W,h,\mathbf{y}) = \mathcal{N}(\alpha; K_\alpha \widehat{\boldsymbol{m}}, K_\alpha)$.

4.2.2 Sampling h given α and W

To sample the values of filter at the inducing times, we need to obtain a formula for

$$p(\boldsymbol{h}|W,\alpha,\mathbf{y}) \propto p(y|W,\alpha,\boldsymbol{h})p(\boldsymbol{h}).$$

Once again, this is a product of Gaussians pdfs, and so the result is a Gaussian pdf as well. The prior for $\boldsymbol{h}$ is straight-forward, since h is a draw from a Gaussian process:

$$p(\boldsymbol{h}) = \mathcal{N}(\boldsymbol{h}; 0, K_h),$$

where K_h is the $N_h \times N_h$ matrix with entries $K_{DEQ}(t_i, t_j)$, $1 \leq i, j \leq N_h$.

To find the distribution of $p(y|W,\alpha,\boldsymbol{h})$ we need to be careful, since $\boldsymbol{h}$ are the values of h at a discrete set of points, and these values need to be estimated, using linear interpolation, from the values of the jumps.

It is motivating to consider the following approach to finding the mean of $p(\mathbf{y}|W,\alpha,\boldsymbol{h})$. If we have a matrix A, so that $\mathbf{y} = A\boldsymbol{h}$, then the least-squares solution of this system of equations coincides with the maximum likelihood estimate of for $p(\mathbf{y}|W,\alpha,\boldsymbol{h})$, which is the mean of this Gaussian distribution. Thus we have

$$\boldsymbol{h} = A^\dagger \mathbf{y},$$

where $A^\dagger$ is the Moore-Penrose pseudo-inverse of A, $A^\dagger = (A^T A)^{-1} A^T$.

To find the i^{th} row matrix A, we observe that $y_i = \sum_j h(t_i - \tau_j) dX_{\tau_j}$. For each time point $t_i - \tau_j$, we find interval $t_l^{(i,j)}, t_r^{(i,j)}$ in the complement of the inducing points with the property that $t_i - \tau_j \in (t_l^{(i,j)}, t_r^{(i,j)})$. Taking

$$a_{i,j} = \frac{(t_i - \tau_j) - t_l^{(i,j)}}{t_r^{(i,j)} - t_l^{(i,j)}},$$

we obtain the linear interpolation of the value $h(t_i - \tau_j)$ from its values at the inducing points $h(t_i - \tau_j) = (1 - a_{i,j}) h(t_l^{(i,j)}) + a_{i,j} h(t_r^{(i,j)})$, and so

$$y_i = \sum_j \Big((1 - a_{i,j}) h(t_l^{(i,j)}) + a_{i,j} h(t_r^{(i,j)}) \Big) X_{\tau_j} \tag{4.2}$$

Thus we obtain an algorithm for finding the coefficients of A: initialize A to an $N_h \times N_h$ matrix of zeros, for each data point (t_i, y_i) and each jump X_{τ_j} and time τ_j, we add $X_{\tau_j}(1 - a_{i,j})$ to the (i,k) entry of A and and $X_{\tau_j} a_{i,j}$ to the $(i, k+1)$ entry of A, where k is the index of in the ordered list of inducing times of $t_l^{(i,j)}$.

Returning to finding the Gaussian distribution for $p(\boldsymbol{y}|W, \alpha, \boldsymbol{h})$. From the analysis above, we have that given a data point (t_i, y_i):

$$p(y_i|\boldsymbol{h}, W, \alpha) = \frac{1}{\sqrt{2\pi}\sigma_{err}} \exp\left[-\frac{1}{2\sigma_{err}^2} \Big(y_i - \sum_{j=1}^{\infty} h(t - \tau_j) dX_{\tau_j} \Big)^2 \right]$$

$$= \frac{1}{\sqrt{2\pi}\sigma_{err}} \exp\left[-\frac{1}{2\sigma_{err}^2} \Big(y_i - \sum_{j=1}^{\infty} \big((1 - \alpha_{i,j}) h(t_l^{(i,j)}) + \alpha_{i,j} h(t_r^{i,j}) \big) dX_{\tau_j} \Big)^2 \right]$$

$$= \frac{1}{\sqrt{2\pi}\sigma_{err}} \exp\left[-\frac{1}{2\sigma_{err}^2} \Big(y_i - \omega_i^T \boldsymbol{h} \Big)^2 \right],$$

where ω_i is the i^{th} row of A.

Thus we are in a position where our previous calculations apply, and we have that

$$p(y_i|W, \alpha, \boldsymbol{h}) = \mathcal{N}(\boldsymbol{h}; \boldsymbol{m}_i, \Sigma_i^{-1}),$$

where $\Sigma_i^{-1} = \frac{1}{\sigma_{err}^2} \omega_i \omega_i^T$, and $\boldsymbol{m}_i' = \Sigma_i y_i \omega_i$.

We obtain the likelihood for the data set as

$$p(\boldsymbol{y}|W,\alpha,\boldsymbol{h}) = \prod_i p(y_i|W,\alpha,h),$$

is a product Gaussian pdfs, and we have that

$$p(\boldsymbol{y}|W,\alpha,\boldsymbol{h}) = \mathcal{N}(\boldsymbol{h};\boldsymbol{m}_h,\Sigma_h^{-1}),$$

where

$$\Sigma_h^{-1} = (\sum_i \Sigma_i)^{-1} \text{ and } \boldsymbol{m}_h = \Sigma_h^{-1}(\sum_i \boldsymbol{m}_i').$$

Putting this together with our prior $p(\boldsymbol{h}|\alpha,W,\boldsymbol{y}) \propto p(\boldsymbol{y}|\boldsymbol{h},W,\alpha)p(\boldsymbol{h})$, and $\boldsymbol{h} \sim \mathcal{N}(\boldsymbol{h};0,K_h)$, we see that

$$p(\boldsymbol{h}|\boldsymbol{y},W,\alpha) \sim \mathcal{N}(\boldsymbol{h};\boldsymbol{m},K),$$

where

$$K = (\Sigma_h + K_h^{-1})^{-1} \text{ and } \quad \boldsymbol{m}_h = K(\Sigma_h \boldsymbol{m}_h)$$

To sample the filter h, we sample its values at inducing points $\boldsymbol{h}$, and then infer the filter from its values at these points.

Algorithm for sampling h given W, α, y

Assume that we are given inducing times $u_1 < u_2 < \cdots < u_{N_h}$, and parameters s, w and σ_h, for the length-scale, window and strength, respectively, of the kernel $\mathcal{K}_{DEQ}(t,t')$. Set $\boldsymbol{u} = (u_1, u_2, \ldots, u_{N_h})$. Suppose that W is approximated by a jump process with M jumps. For each jump dW_{τ_j} of W, let $X_{\tau_j} = \alpha_j \sqrt{W_{\tau_j}}$.

A. Obtain the distribution for the inducing values $\boldsymbol{h} = (h_1, \ldots, h_{N_h})$, at the inducing times $(u_1, \ldots, u_{N_h})$:

   1. Compute the prior covariance matrix K_h whose (i,j) entry is $K_{DEQ}(u_i, u_j)$, and let $\Sigma = K_h^{-1}$. Initialize $\boldsymbol{m}_h = 0 \in \mathbb{R}^{N_h}$.

   For each data point (t_i, y_i), initialize $\omega^i = 0 \in \mathbb{R}^{N_h}$, and do the following:

   2. For each $j = 1, 2, \ldots, M$, if $u_0 < t_i - \nu_j < t_{N_h}$, do the following:
      a. Let $(u_1^{i,j}, u_2^{i,j})$ be the interval in the complement of the inducing times that contains $t_i - \nu_j$, and let k_1 be the index of $\boldsymbol{u}$ corresponding to $u_1^{i,j}$.
      b. Let $D_{i,j} = X_{\tau_j}(t_i - \tau_j - u_1^{i,j})/(u_2^{i,j} - u_1^{i,j})$

c. Update the entries of ω^i:

$$\omega^i_{k_1} \leftarrow \omega^i_{k_1} + X_{\tau_j} - D_{i,j}$$

$$\omega^i_{k_1+1} \leftarrow \omega^i_{k_2} + D_{i,j}$$

3. Let $\hat{\omega} = \omega_i/\sigma_{err}$, and update the covariance matrix and $\boldsymbol{m}_h$:

$$K_h \leftarrow K_h - \frac{1}{1+\hat{\omega}^T K_h \hat{\omega}} K_h \hat{\omega}\hat{\omega}^T K_h$$

$$\boldsymbol{m}_h \leftarrow \boldsymbol{m}_h + \frac{y_i}{\sigma^2_{err}}\hat{\omega}.$$

We have a distribution for $\boldsymbol{h}$: $p(\boldsymbol{h}|W,\alpha,\mathbf{y}) = \mathcal{N}(\boldsymbol{h}; K_h\boldsymbol{m}_h, K_h)$.

B. Sample $\boldsymbol{h}$ from the distribution we have just found, and sample h from the posterior Gaussian process using GP regession given data points $(u_1,h_1),\ldots,(u_{N_h},h_{N_h})$.

4.2.3 Sampling from $p(W|h,\mathbf{y})$ using Metropolis-Hastings in Gibbs Sampling

In this section, we describe how we sample paths of the subordinator process from $p(W|h,\mathbf{y})$.

We make use of a Metropolis-Hastings in Gibbs method employed in Kındap and Godsill (2022b), in which we successively sample the subordinator process in small intervals, taken from a partition of the domain of the subordinator process.

Suppose that $I=[0,T]$ is the domain of the subordinator process. Let $\mathscr{I} = \{I_0, I_1, \ldots I_{k-1}\}$ be a partition of the interval I into subintervals I_i for $i \in \{0,\ldots,k-1\}\}$. We let J_j denote the set of all jumps of the subordinator that occur in the interval I_j and J_{-j} denote the collection of jumps that occur outside of the interval I_j. We sample approximately from the posterior $p(W|h,\mathbf{y},\alpha)$, by sampling successively from $p(W|J_{-j},h,\mathbf{y},\alpha)$, where $j = n \bmod(k)$ for $n=0,1,2,\ldots$. While this improves the convergence of the sampler, it necessitates sampling from distribution of sample path s of the subordinator given J_{-j},h,α and $\mathbf{y}$, which we do by using a Metropolis-Hastings step.

Assume that we sample the jumps of the subordinator in the interval I_i during the $n+1$ iterate of the Gibbs sampler. We let $W^{(n)}$ denote the previously accepted subordinator, and we let W' denote the new, proposed, subordinator, which has the same jumps as $W^{(n)}$ outside of I_i, but with jumps J_i' in the interval I_i. Our proposal distribution for the MH step is the distribution of the new choice of the subordinator given the previous choice. This has density $p(W'|W^{(n)})$. Recall that by Bayes' Theorem, we have that $p(W'|J_{-i},h,\mathbf{y}) \propto p(\mathbf{y}|W',h)p(W')$.

Notice that since W' agrees with J_{-i} outside of I_i, the subordinator with the jumps of W' and J_{-i} is the same as W'.

Now, we have that the acceptance probability is given by

$$a(J_i', W^{(n)}) = \min\left(1, \frac{p(W'|J_{-i}, \boldsymbol{y}, h)p(W^{(n)}|W')}{p(W^{(n)}|J_{-(i-1)}, \boldsymbol{y}, h)p(W'|W^{(n)})}\right) \tag{4.3}$$

$$= \min\left(1, \frac{p(\boldsymbol{y}|W', h)}{p(\boldsymbol{y}|W^{(n)}, h)}\right).$$

Notice that the quotient:

$$\frac{p(W'|J_{-i}, \boldsymbol{y}, h)p(W^{(n)}|W')}{p(W^{(n)}|J_{-(i-1)}, \boldsymbol{y}, h)p(W'|W^{(n)})} = \frac{p(\boldsymbol{y}|W', h)p(W')p(W^{(n)}|W')}{p(\boldsymbol{y}|W^{(n)}, h)p(W^{(n)})p(W'|W^{(n)})}$$

$$= \frac{p(\boldsymbol{y}|W', h)p(W', W^{(n)})}{p(\boldsymbol{y}|W^{(n)}, h)p(W', W^{(n)})} = \frac{p(\boldsymbol{y}|W', h)}{p(\boldsymbol{y}|W^{(n)}, h)}.$$

In summary, to approximate samples of $p(W|h, \boldsymbol{y})$, we can successively sample from $p(W|J_{-i}, h, \boldsymbol{y})$ by sampling from $p(W|J_{-i})$, and accepting these samples with probability $\min(1, p(\boldsymbol{y}|W', h)/p(\boldsymbol{y}|W^{(n)}, h))$.

Algorithm for sampling the subordinator

Assume that we are given an initial subordinator W, a filter h, data points $\boldsymbol{y}$ and a partition $\mathscr{I} = \{I_i\}_{i=0}^{k-1}$ of the interval.

- For $i = 0, 1, \ldots, k-1$ do the following:

  1. Let J_{-i} denote the jumps of W in the complement of I_i
  2. Sample the jumps J_i' of a subordinator process with intensity the length of I_i in the interval I_i.
  3. Let W' be the subordinator process with jumps J_i' in I_i and jumps J_{-i} outside i.
  4. compute the acceptance probability $a(J_i'|W)$ using (4.3). See page 35 for the algorithm.
  5. Sample u uniformly in $[0, 1]$, and accept W' if $u < a(J', W)$ (that is, set $W = W'$) otherwise reject W'.

4.2.4 The Gibbs Sampling Algorithm

Assume that we are given a collection of data points $\boldsymbol{y} = \{t_i, y_i\}$ that we wish to model. Fix a number of times n_0 that we wish to sample from each marginal distribution.

1. Initialization.

    a. Initlialize parameters δ, γ, λ for the subordinator GIG process, and the window, scale parameters of the covariance function of the filter.

    b. Choose a partition $\mathscr{I} = \{I_i\}_{i=0}^{k-1}$ of the interval into k subintervals I_i.

    c. Initialize an approximate subordinator process $\{W(\tau) : \tau \geq 0\}$, for each jump of W. For each jump dW_{τ_j} of the subordinator, sample $\alpha_j \sim \mathscr{N}(0,1)$, and sample a filter from $GP(0, K_{DEQ}(t_1, t_2))$.

Repeat steps 2, 3 and 4 n_0 times:

2. Use the algorithm on page 4.2.3 for sampling a subordinator process W from $p(W|h, \boldsymbol{y})$ using the partition $\mathscr{I}$.

3. Sample α from $p(\alpha|W, h, \boldsymbol{y})$, see page 36.

4. Sample h using algorithm given on page 4.2.2.

4.3 Extensions to the sampler

For modelling real-world data, the Gibbs sampler as presented has some serious deficiencies. In this section, we present some improvements to the sampler, which address some of these problems.

4.3.1 Sampling the position of the inducing times.

Let us describe a simple method to allow the sampler to attempt to find better inducing times. Instead of sampling immediately from $p(\boldsymbol{h}|W, \alpha, y)$ with the times of $\boldsymbol{h}$ fixed. We add an additional step where we sample a new sequence of inducing times. More precisely, we wish to draw a sample from the joint distribution $p(\boldsymbol{h}_t, \boldsymbol{h}|W, \alpha, \boldsymbol{y})$, which again we can accomplish by introducing a Metropolis-Hastings step into the Gibbs sampler.

We assume that the positions of the inducing times are uniformly distributed in a subinterval of $\mathbb{R}$ outside of which the filter nearly vanishes. In practice, we can determine suitable values for these end points from the covariance matrix of the filter. We fix inducing times

$t_1 < t_{N_h}$ at the end points of the interval, and initialize the inducing times at evenly spaced points in $[t_1, t_{N_h}]$. In each iteration of the Gibbs sampler, we sample t_i uniformly in the interval (t_{i-1}, t_{i+1}) for $i = 1, 2, \ldots, N_n - 1$.

We sample the inducing times $\boldsymbol{h}'_t$ and values $\boldsymbol{h}'$ from $p(\boldsymbol{h}'_t, \boldsymbol{h}' | \boldsymbol{h}_t, \boldsymbol{h})$, and accept with probability

$$a = \min\left(1, \frac{p(\boldsymbol{y}|\boldsymbol{h}', \boldsymbol{h}'_t, W, \alpha)}{p(\boldsymbol{y}|\boldsymbol{h}, \boldsymbol{h}_t, W, \alpha)}\right).$$

Notice that this is the correct acceptance probabilty since

$$\frac{p(\boldsymbol{y}|\boldsymbol{h}', \boldsymbol{h}'_t, W, \alpha)}{p(\boldsymbol{y}|\boldsymbol{h}, \boldsymbol{h}_t, W, \alpha)} = \frac{p(\boldsymbol{y}|\boldsymbol{h}', \boldsymbol{h}'_t, W, \alpha) p(\boldsymbol{h}_t, \boldsymbol{h} | \boldsymbol{h}'_t, \boldsymbol{h}') p(\boldsymbol{h}'_t, \boldsymbol{h}')}{p(\boldsymbol{y}|\boldsymbol{h}, \boldsymbol{h}_t, W, \alpha) p(\boldsymbol{h}'_t, \boldsymbol{h}' | \boldsymbol{h}_t, \boldsymbol{h}) p(\boldsymbol{h}_t, \boldsymbol{h})}.$$

$$\frac{p(\boldsymbol{h}'_t, \boldsymbol{h}' | W, \alpha, \boldsymbol{y}) p(\boldsymbol{h}_t, \boldsymbol{h} | \boldsymbol{h}'_t, \boldsymbol{h}_t)}{p(\boldsymbol{h}_t, \boldsymbol{h} | W, \alpha, \boldsymbol{y}) p(\boldsymbol{h}'_t, \boldsymbol{h}' | \boldsymbol{h}_t, \boldsymbol{h})}.$$

4.3.2 Learning the noise and the parameters of the filter

Similar ideas to those used to optimize the hyper-parameters, *e.g.* the length scale, of Gaussian processes can be used in these models to estimate the hyper-parameters for the underlying filter in the GPCM.

Let θ be either the length-scale, window or strength of the covariance function $\mathscr{K}_{DEQ}$ of the filter. We will optimize our model with respect to θ by increasing the marginal likelihood $p(y|W, \alpha, h_t, \boldsymbol{h}, \theta)$ using gradient ascent.

We have that $p(y|W, \alpha, h_t, \boldsymbol{h}, \theta)$ is a Gaussian, and we have

$$\boldsymbol{y} = C\boldsymbol{h} + \varepsilon_i, \quad \text{where } \varepsilon \sim \mathscr{N}(0, \sigma^2_{err}\mathbb{I}),$$

where the i^{th} row of C is given by (4.2).

Now, we have that $\boldsymbol{h}$ is sampled from an Gaussian process, we have that

$$p(\boldsymbol{y}|W, \alpha, \boldsymbol{h}) = \mathscr{N}(\boldsymbol{h}; \boldsymbol{m}_h, \Sigma_h^{-1}).$$

Thus we have

$$\boldsymbol{y} \sim \mathscr{N}(\boldsymbol{y}; C\boldsymbol{m}_h, C\Sigma_h^{-1}C^T + \sigma^2_{err}\mathbb{I}).$$

We have that

$$\frac{\partial}{\partial\theta}\log p(\boldsymbol{y}|W, \alpha, \boldsymbol{h}, h_t) = -\frac{\partial}{\partial\theta}\left[\log(|C\Sigma_h^{-1}C^T|) - \frac{1}{2}(\boldsymbol{y} - C\boldsymbol{m}_h)^T(C\Sigma_h^{-1}C + \sigma^2_{\text{err}}\mathbb{I})(\boldsymbol{y} - C\boldsymbol{m}_h)\right]$$

$$= -\text{trace}\left((C\Sigma_h^{-1}C)^{-1}\frac{\partial}{\partial\theta}C\Sigma^{-1}C^T \right) + \frac{1}{2}(\boldsymbol{y}-C\boldsymbol{m}_h)^T C\frac{\partial\Sigma^{-1}}{\partial\theta}C(\boldsymbol{y}-C\boldsymbol{m}_h)$$

$$= \frac{1}{2}\text{trace}\left(\left(\boldsymbol{v}\boldsymbol{v}^T - (C\Sigma^{-1}C^T)^{-1}\right)C\frac{\partial\Sigma^{-1}}{\partial\theta}C^T \right) \quad \text{where } \boldsymbol{v} = (C\Sigma^{-1}C^T)^{-1}(\boldsymbol{y}-C\boldsymbol{m}_h),$$

and $\frac{\partial\Sigma^{-1}}{\partial\theta}$ denotes the matrix whose i,j entry is the derivatives with respect to θ of the i,j entry of Σ^{-1}.

We have seen that we can compute each of the terms in this formula. Hence, after fixing a learning rate $\eta > 0$, we can update θ to θ' according to

$$\theta' = \theta + \eta\frac{\partial}{\partial\theta}\log p(\boldsymbol{y}|W,\alpha,\boldsymbol{h},h_t).$$

Similarly, using the fact that each $y_i \sim \mathcal{N}(f(t_i),\sigma_{err}^2)$, we can optimize our model with respect ot σ_{err}^2 using gradient ascent along the level set of the likelihood $p(\boldsymbol{y}|W,h)$.

4.4 Evaluating the sampler

In this section we will consider data $\boldsymbol{y} = \{(t_i,y_i)\}$ that is generated from a given Gaussian process convolution model obtained from a given filter h_0, subordinator W_0 and GH process X_0, and evaluate the ability of the Gibbs sampler to reconstruct the constituent parts of the GPCM from the data. Throughout this section we will assume that the strength of the filter $\sigma_h = 1$.

4.4.1 Recovering the components of a GPCM

Let us demonstrate to what extent our Gibbs sampler can recover the subordinator process, the generalized hyperbolic process and the filter of a GPCM, and how well the resulting GPCM formed using these components models data from the original process.

Constructing the example

We generated a sample path $W = \{W(\tau) : \tau \geq -3\}$ from a GIG distribution with parameters $\delta = 1, \gamma = 0.1$ and $\lambda = -0.6$, Figure 4.1. Then for each jump dW_{τ_i} in the shot-noise approximation of W, we drew $\alpha_i \sim \mathcal{N}(0,1)$, and formed an approximation of a GH process $X = \{X(\tau) : \tau \geq -3\}$, with jumps dX_{τ_i}, with subordinator W by setting $dX_{\tau_i} = \alpha_i\sqrt{W_{\tau_i}}$, see Figure 4.2. Next we drew a sample of a Gaussian process $h \sim GP(0,\mathcal{K}_{DEQ}(t_1,t_2))$,

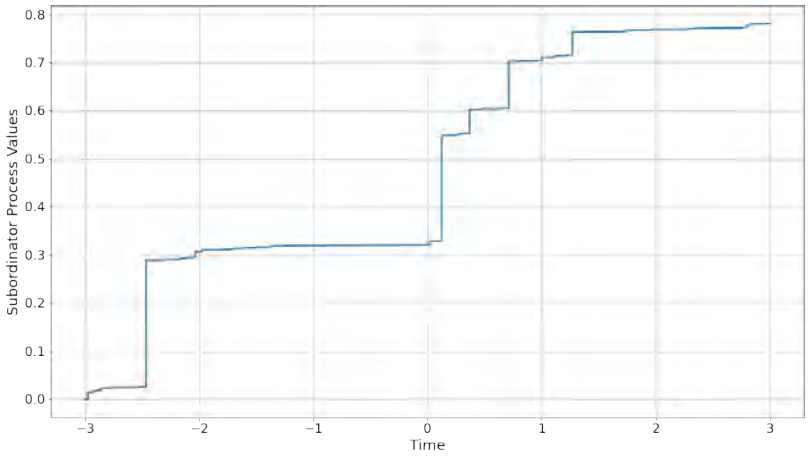


Fig. 4.1 Subordinate GIG process with $\delta = 1, \gamma = 0.1$ and $\lambda = -0.6$

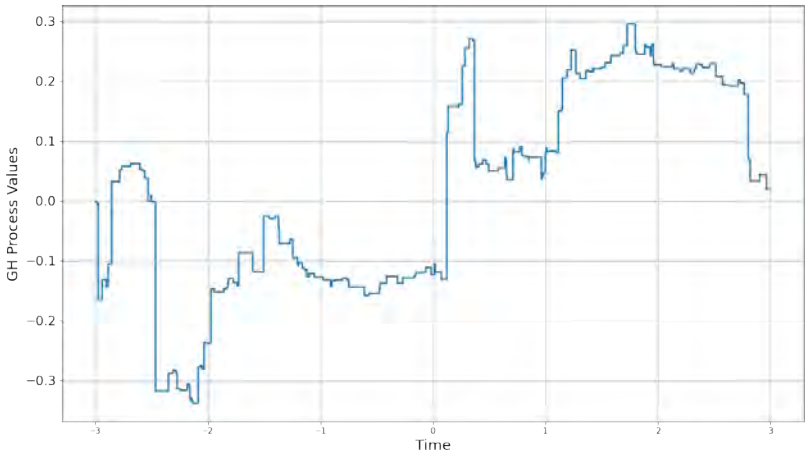


Fig. 4.2 GH Process with jumps $dX_{\tau_i} = \alpha_i \sqrt{W_{\tau_i}}$, where $\alpha_i \sim \mathcal{N}(0,1)$ with subordinator from Figure 4.1.

with window and scale set to 0.4 and 0.15, respectively, Figure 4.3, and finally formed the convolution

$$f(t) = \int_{-\infty}^{\infty} h(t-\tau)dX(\tau) = \sum_j h(t-\tau_j)dX_{\tau_j},$$

see Figure 4.4

We sampled 250 points $\{t_i\}_{i=1}^{250}$ from the interval $[-3,3]$, and formed a data set $\boldsymbol{y} = \{(t_i, f(t_i)\}_{i=1}^{250}$. For this first experiment we analyzed how well the sampler recovers the subordinator, the GH process and the filter from the data. We held the parameters of the Lévy processes and the filter fixed to be the same as those used to generate the data. We will also fix the inducing times to be located on an evenly spaced set of of points.

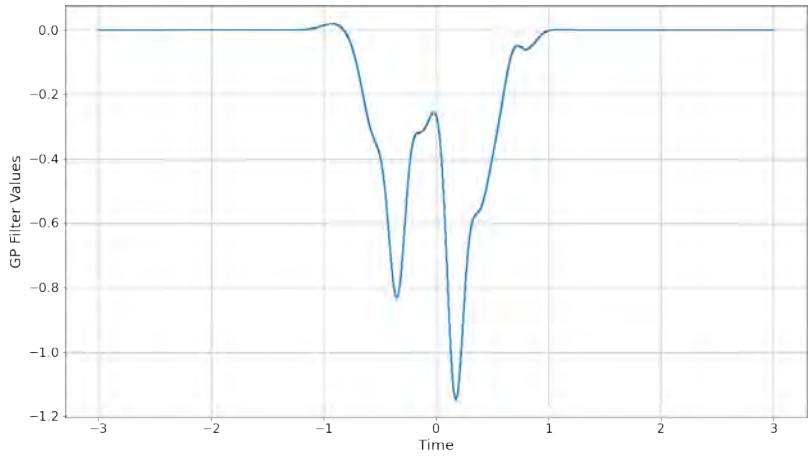


Fig. 4.3 A Gaussian process $h \sim GP(0, \mathscr{K}_{DEQ}(t_1, t_2))$ with window=0.4 and scale=0.1.

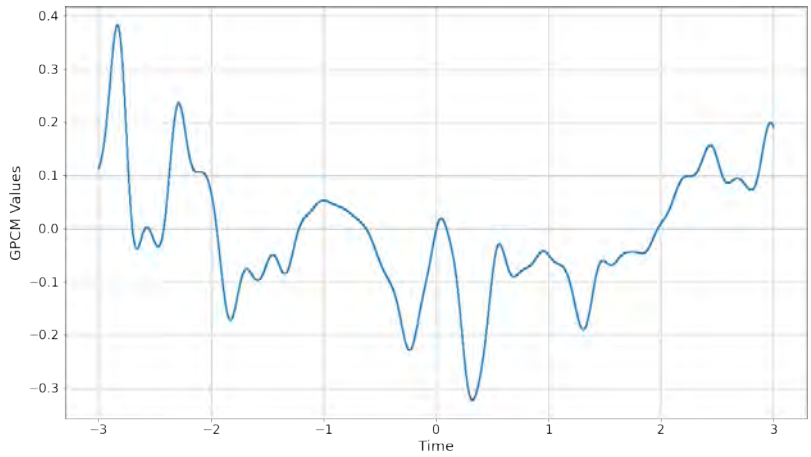


Fig. 4.4 Convolution of the Gaussian process h from Figure 4.3 with the GH process X from Figure 4.2.

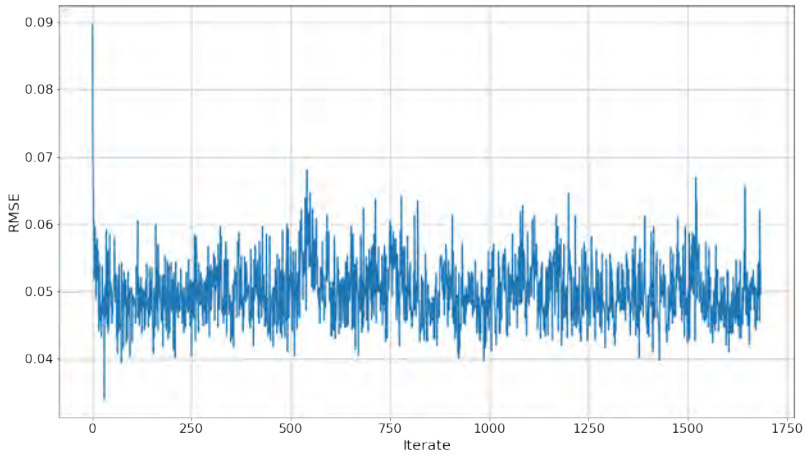


Fig. 4.5 RMSEs for the Gibbs sampler.

One feature of these models, is that it is possible to multiply the filter by some constant, while dividing the subordinator by the same constant an obtain the same convolution, as a result we don't don't necessarily expect the filter from Gibbs sampler to be have the same "height" as the target filter h_0. To compare them, we fix the width of the range of the inducing values (the length of the smallest interval containing all of the inducing values). Later we will investigate how this effects the efficacy of the model.

The root mean squared errors suggest that the sampler may converge quickly, Figure 4.5. We took a burn in time of 700 iterations of the sampler, and did not observe substantial differences in the results when I varied the gap between selected samples.

Recovering the GH process

We find that the sample paths of the GH process X obtained by the Gibbs sampler seem to be exploring the space of paths quite well, Figure 4.6. Moreover, comparing the sampled paths with the target path, the picture shows that the sampler is drawing processes with large jumps in reasonable places with reasonable sizes. On average, we see that the shape of the sample paths drawn by the sampler is similar to the shape of the target process, Figure 4.7.

Recovering the subordinator

The multiplication of each jump of the subordinator by some α_i makes it possible for the scale of the sizes jumps of the samples to be different from that of jumps of the the target subordinator. Indeed we observed that in this example; however, viewing the paths of samples of the subordinator process, we see that the Gibbs sampler is behaving well, Figure 4.8.

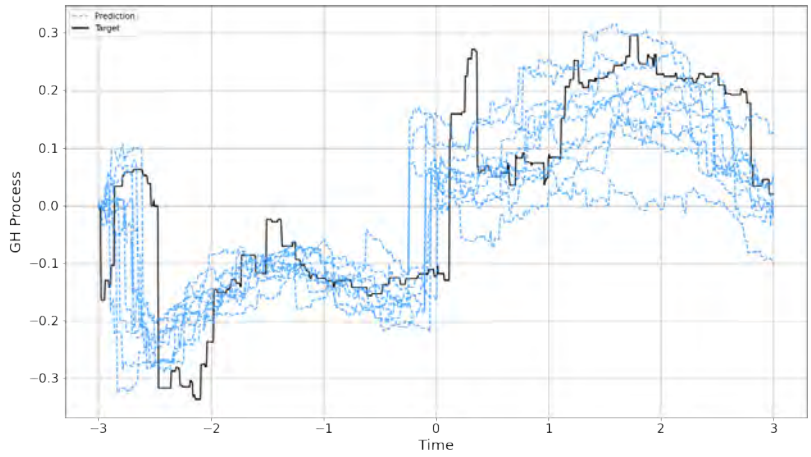


Fig. 4.6 Sample paths of the GH process obtained by the Gibbs sampler.

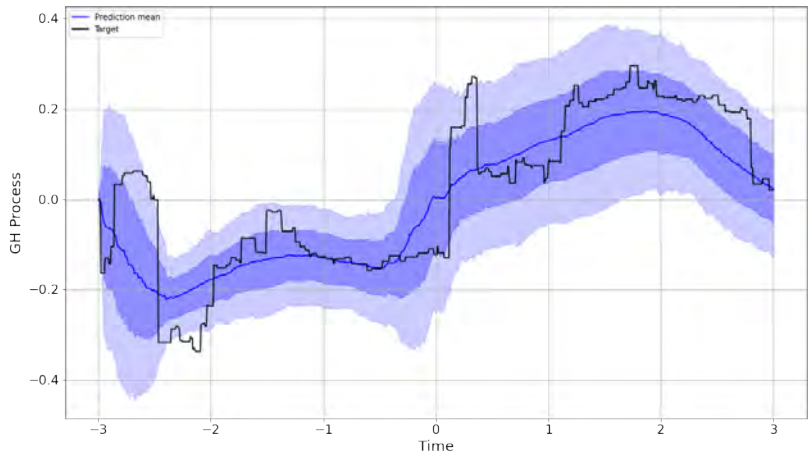


Fig. 4.7 The mean and error bars for the average values of the GH processes found by the sampler. Error bars are one and two standard deviations.

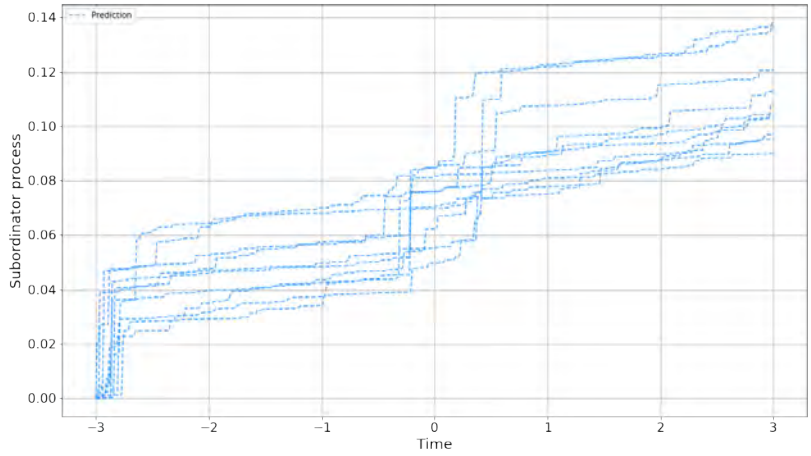


Fig. 4.8 Paths of samples of the subordinator process.

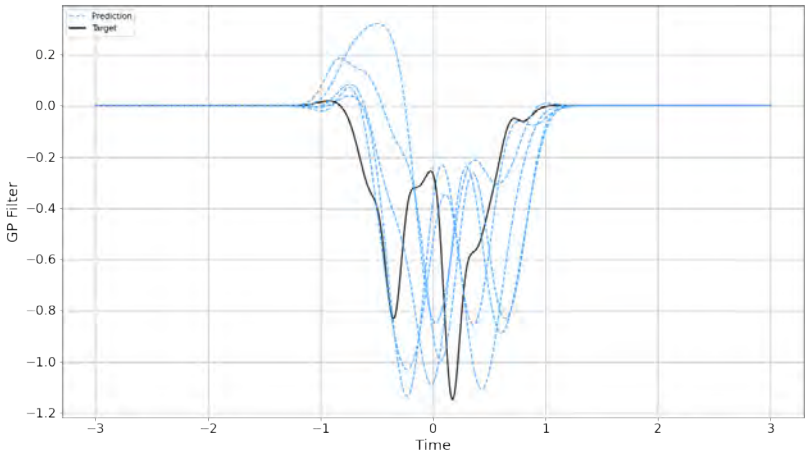


Fig. 4.9 Samples of the filter

Recovering the filter

In addition to the scaling ambiguity that we mentioned above, there is also a flip ambiguity caused by multiplying both the filter and the GH process, or more precisely the α_i by -1. This can cause the average values of the filter to be close to zero. We avoided this problem in this example by normalizing our filters to satisfy $h(0) < 0$. We display some of the sampled filters in Figure 4.9, and the average value of the sampled filters in Figure 4.10. We observe that there distribution of the filters is quite broad.

We can also gain insight into how well the sampler is learning the filter from the power spectral densities of the sampled filters. We see in Figure 4.11 that the sample mean of the filters recovers the first two peaks of the spectrum, but that it does not recover the higher frequency peak, which is perhaps too much like noise.

Recovering the convolution

Combining samples of the subordinator, the coefficients α_i and the filter, we obtain samples of the GPCM. We see that the result models the data well, Figure 4.12. However it does not capture high frequency information from the spectrum, Figure 4.13, and so the result is a somewhat too smooth. It also does not capture all of the peaks present in the original model.

Discussion

Let us briefly summarize some of the issues that we observed in this first example, and some questions that need to be addressed.

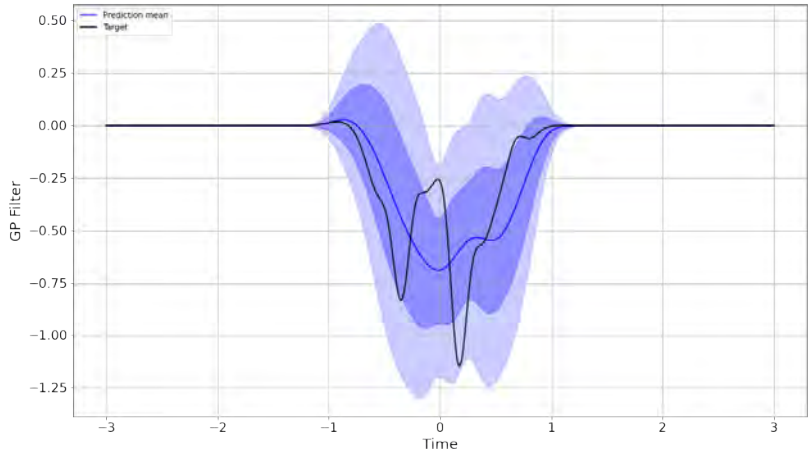


Fig. 4.10 Average values of the sampled filters. Error bars are at one and two standard deviations.

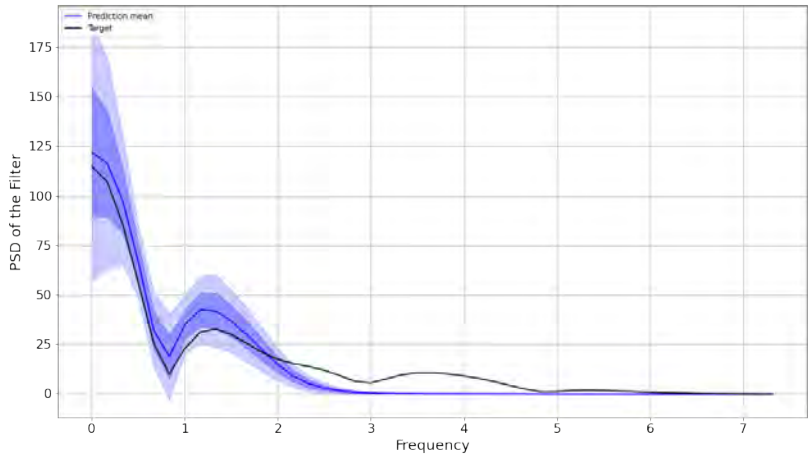


Fig. 4.11 Average power spectral density of the sampled filters. Error bars at one and two standard deviations.

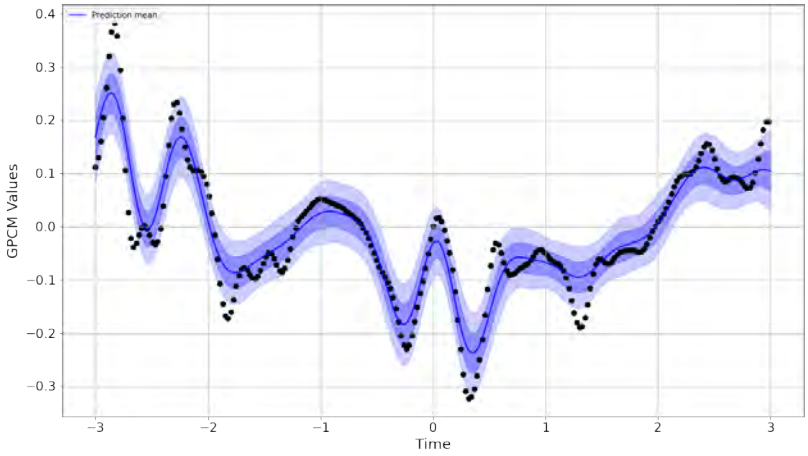


Fig. 4.12 Mean of the sampled convolutions $f(t) = \sum_j \alpha_j h(t - \tau_j) dW_{\tau_j}$. Error bars at one and two standard deviations.

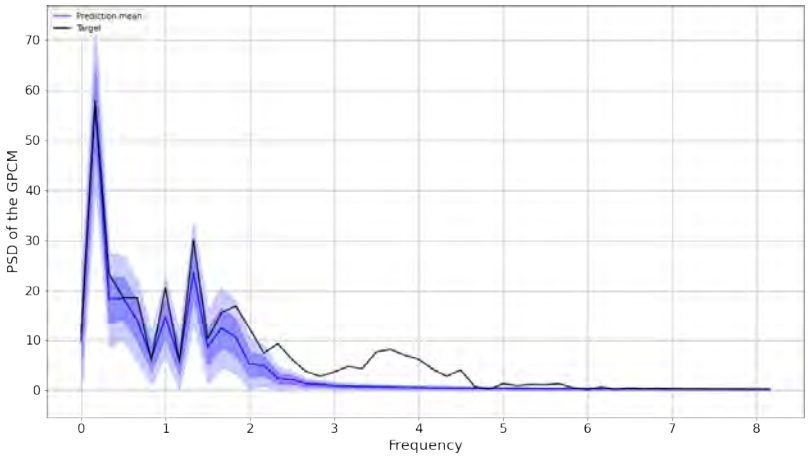


Fig. 4.13 Power spectral desity of the resulting Gaussian process convolution model. Error bars at one and two standard deviations.

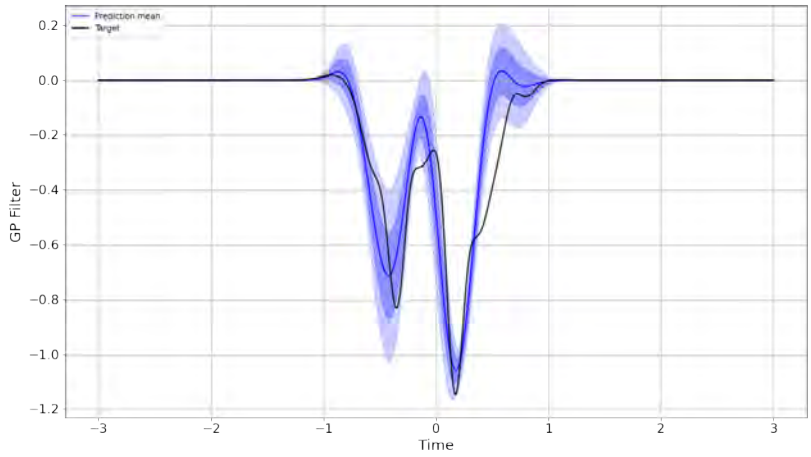


Fig. 4.14 Average value of the sampled filter when we allow the model to sample the inducing times. Error bars at one and two standard deviations.

- The variance of the samples of the filter is large enough that it is probably not reasonable to suggest that the model learned the filter. One obvious way to improve the sampling of the filter is to allow the model to infer the best inducing times. We will investigate this.
- We decided to scale the inducing points of the filter to fix the width of its range. What happens if we do not do this and what are the effects of this on the final model?
- How well does the model learn the filter, processes and convolution when the data is noisy?
- How does the model cope with missing data?

4.4.2 Sampling the inducing times

To evaluate the effect of allowing the model to sample the inducing times, we leave all other aspects of the previous example alone, but introduce the additional inducing times sampling step into our algorithm.

When we sampled to the inducing times as well as the values at the inducing points, we found that the sampler was able to learn a filter much more precisely than when we fixed the inducing times Figure 4.14. Sampling the inducing times also resulted in a slightly more accurate final model: the RMSE of the sample mean improved from 0.0499 to 0.0474.

We note that the average of the samples does not capture the finer structure that is present in the original filter, and indeed this is still the case if we look at examples of the sampled filters, Figure 4.15.

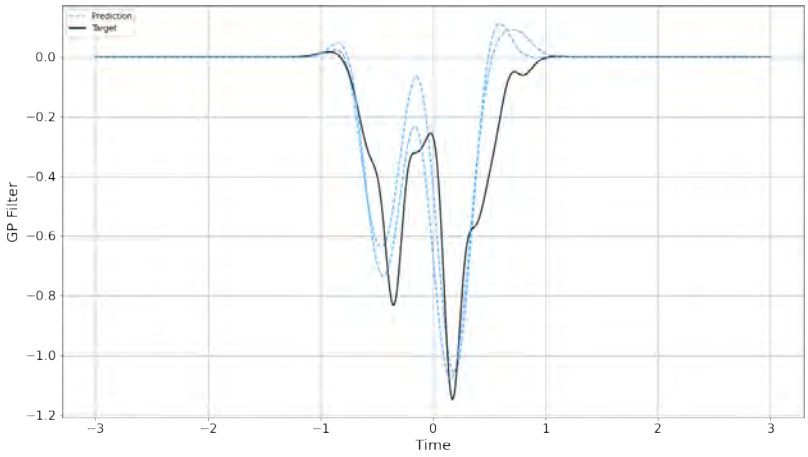


Fig. 4.15 Samples of the filter when we allow the model to sample the inducing times.

In all of the remaining examples, unless otherwise stated, we will use the model which additionally samples the inducing times.

4.4.3 Normalizing the filter

We find that while normalizing the filter (*e.g.* so that its range has has a definite size) makes it possible to visually inspect for whether the sampler is learning the filter and GH process, and it may appear to lead to faster convergence of the sampler, it does not seem to produce better results overall, and in further experiments we will not normalize the filter. We do not claim that this is a deep observation, but it we will take advantage it to present a second in depth example of a GPCM.

We consider a GPCM with subordinator, Figure 4.16, GH process, Figure 4.17, filter, Figure 4.18, which produces the convolution in Figure 4.19.

Comparing the RMSEs for the two models, we see that the errors when we do not normalize the size of the filter are typically smaller, Figure 4.20.

We also observe an improvement in the ability of the filter to learn the spectrum of the GPCM when we do not normalize the filter. The statistics for the samples of the normalized filter are shown in Figure 4.21, and those for the unnormalized filter in Figure 4.22.

As before, we observe that when we do not normalize the filter, the algorithm explores the spaces of filters, Figure 4.23, GIG processes, Figure 4.24, and GH processes, Figure 4.25, well and seems to be generating samples that are close to the target objects., Finally, the result of the samples produces a reasonable model of the data, Figure 4.26.

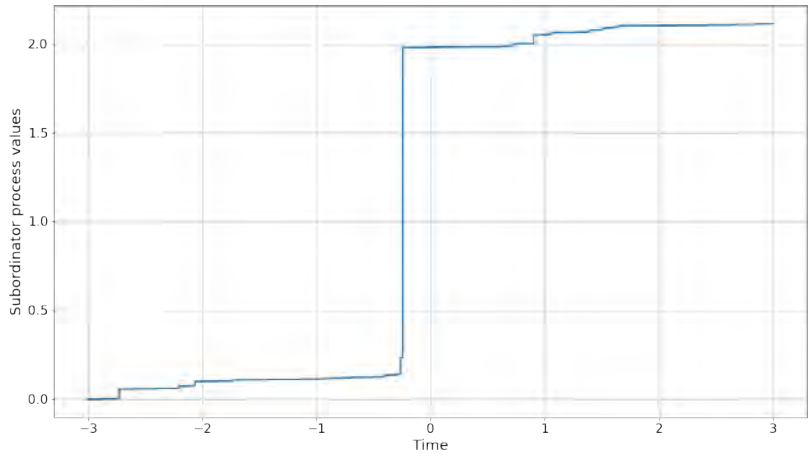


Fig. 4.16 Subordinator process, parameters $\delta = 1.5, \gamma = 0.5$ and $\lambda = -1$.

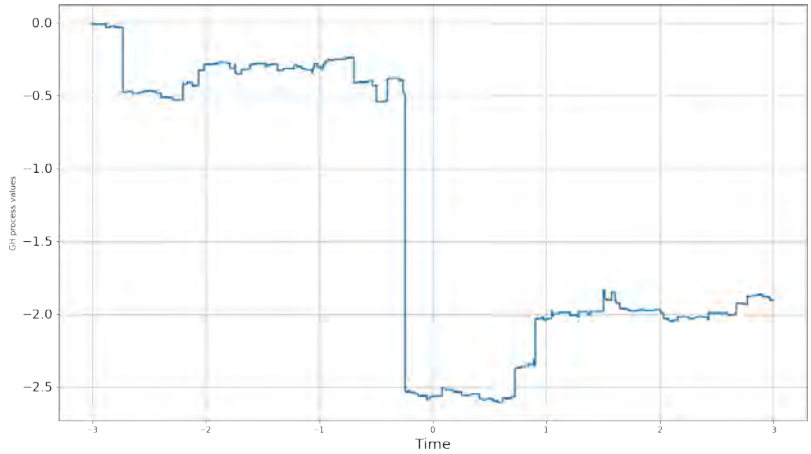


Fig. 4.17 GH process with subordinator from Figure 4.16

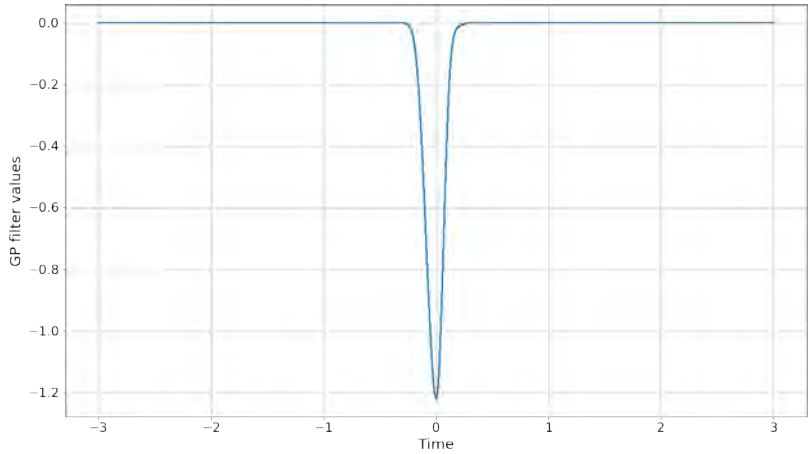


Fig. 4.18 $h \sim GP(0, \mathscr{K}_{DEQ}(t_1, t_2))$

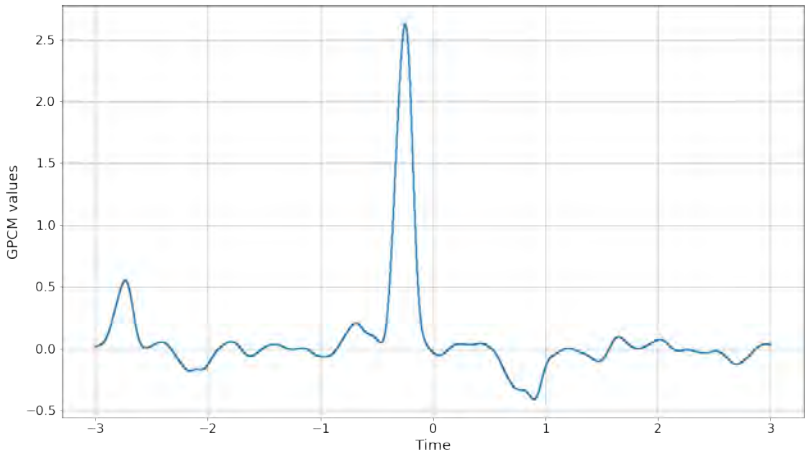


Fig. 4.19 Convolution of the filter from Figure 4.18 and the GH process from Figure **??**.

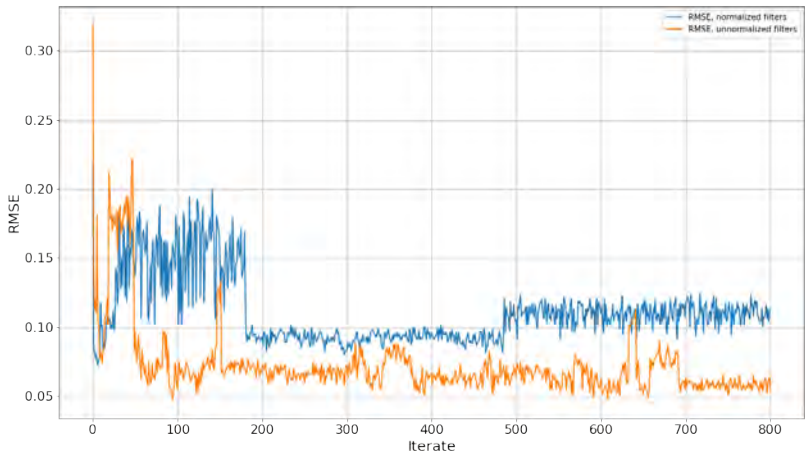


Fig. 4.20 RMSEs for the sampled GPCMs with normalized filters (blue) and unnormalized filters (orange).

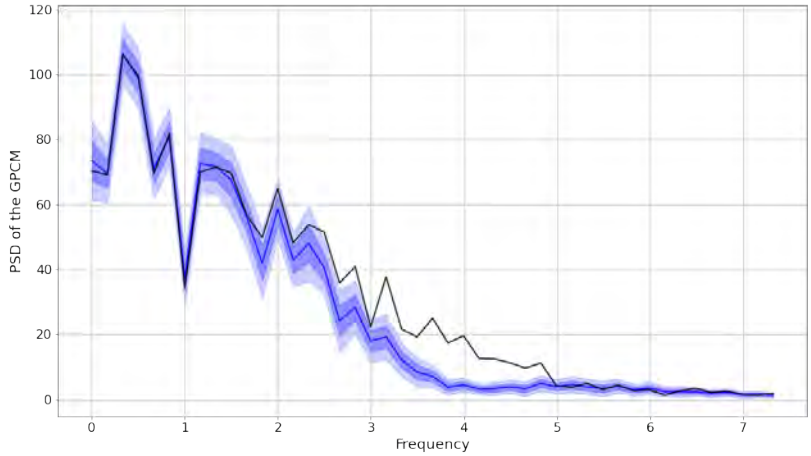


Fig. 4.21 Spectrum of the gpcm when normalized samples of the filter are used. Error bars are at one and two standard deviations.

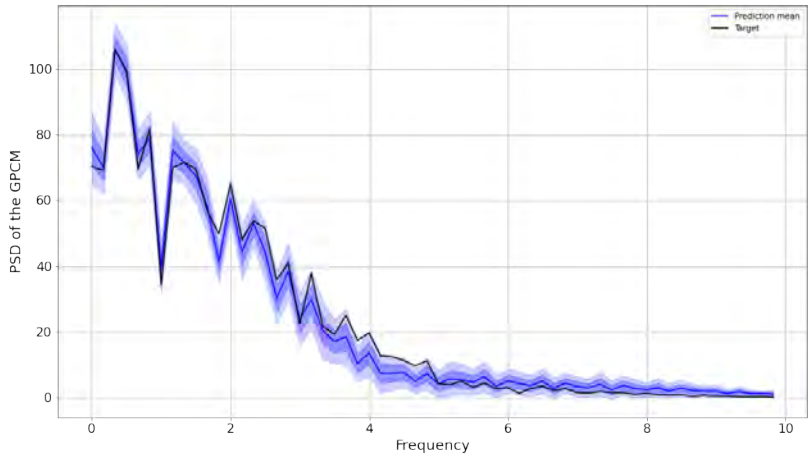


Fig. 4.22 Spectrum of the gpcm when we do not normalize the samples of the filter. Error bars are at one and two standard deviations.

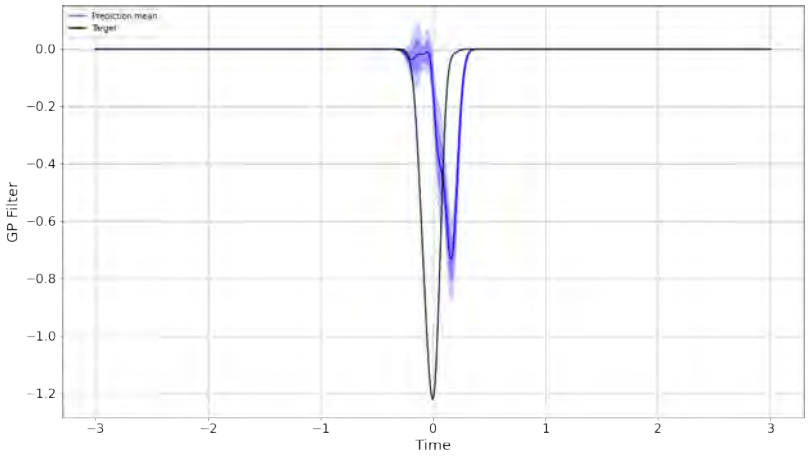


Fig. 4.23 Mean of samples of the unnormalized filter. Error bars are at one and two standard deviations.

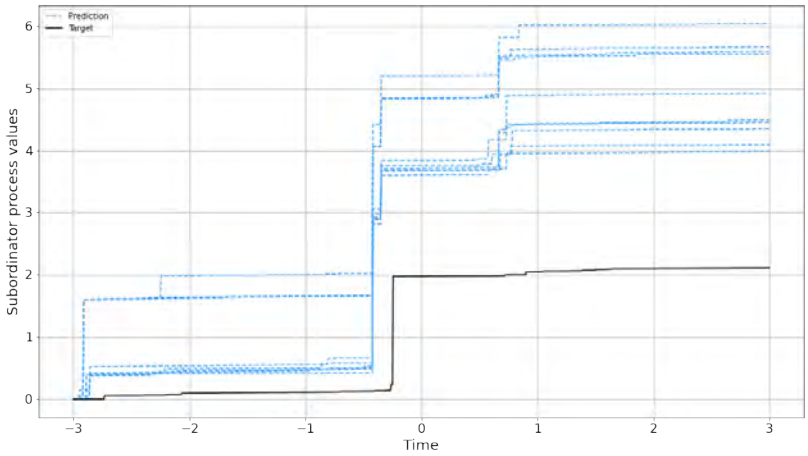


Fig. 4.24 Samples of the GIG process when the filter is not normalized.

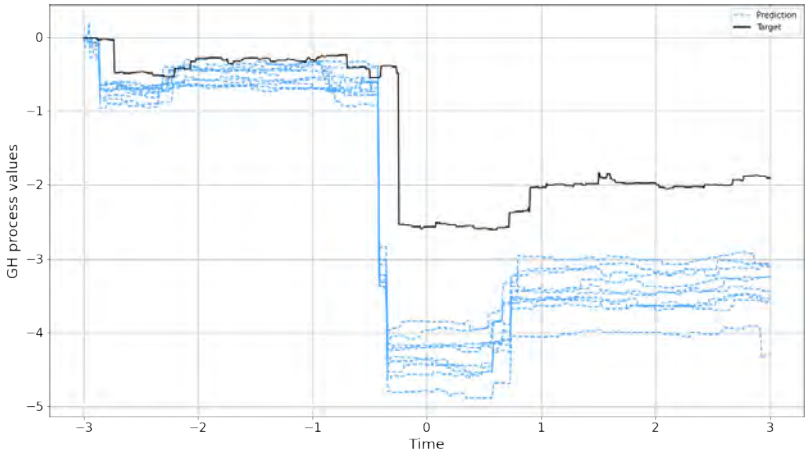


Fig. 4.25 Samples of the GH process when the filter is not normalized.

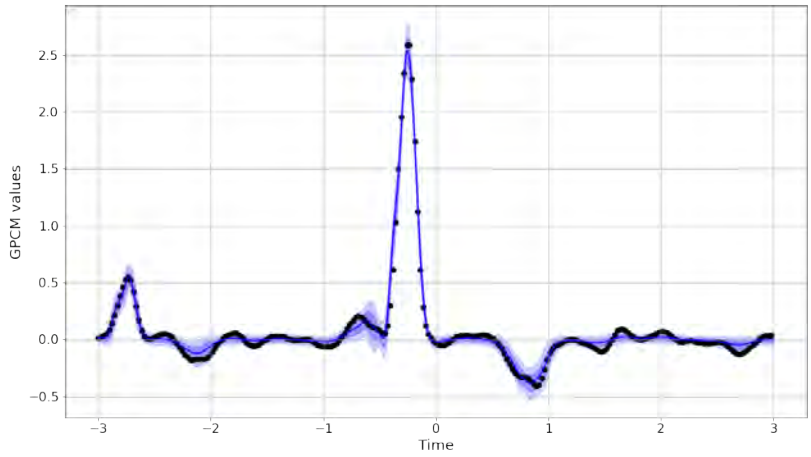


Fig. 4.26 Mean of the resulting Gaussian process convolution model when the samples of the filter are not normalized. Error bars at one and two standard deviations.

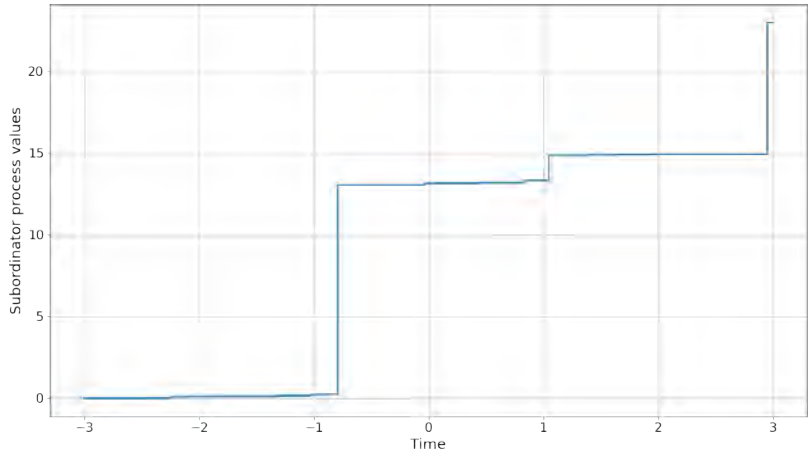


Fig. 4.27 GIG subordinator with parameters $\delta = 1, \gamma = 0.2$ and $\lambda = 0.3$.

4.4.4 Noisy data missing data points

Let us consider a GPCM where we introduce noise into the data, and exclude a subset of the data consisting of half the data points from the training. To this end, we consider a GIG subordinator with parameters $\delta = 1, \gamma = 0.2$ and $\lambda = 0.3$, Figure 4.27 and we consider a filter $h \sim GP(0, \mathscr{K}_{DEQ}(t_1, t_2)$ with scale = 0.2 and window = 0.5, Figure 4.28. We show the GH process $X = \{X_\tau : \tau \geq 0\}$ used in this example in Figure 4.29, and the resulting convolution $f(t) = \sum_j h(t - \tau_j) dX_{\tau_j}$, in Figure 4.30.

We sample 334 data points $y_i = f(t_i) + \varepsilon_i$, $\varepsilon_i \sim \mathscr{N}(0,1)$. We randomly select half of them for the training set $\boldsymbol{y}$ and withhold the remaining half to test the efficacy of the sampler. Let us note that in the previous examples always used 250 points for training. We show the

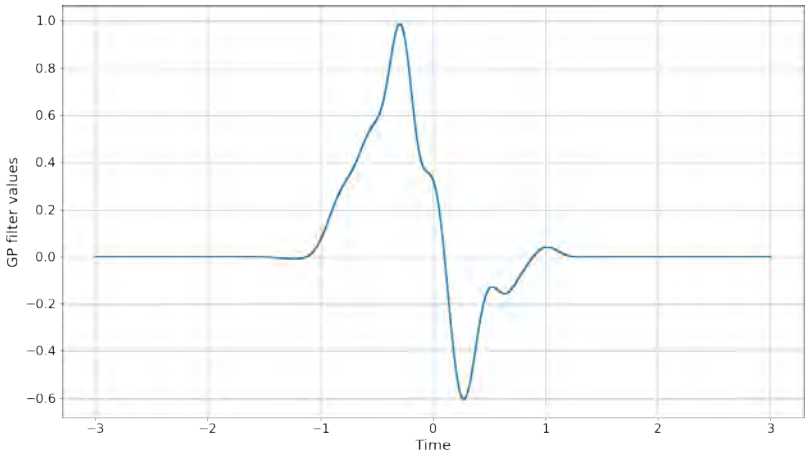


Fig. 4.28 $h \sim GP(0, \mathscr{K}_{DEQ}(t_1, t_2)$ with scale = 0.2 and window = 0.5.

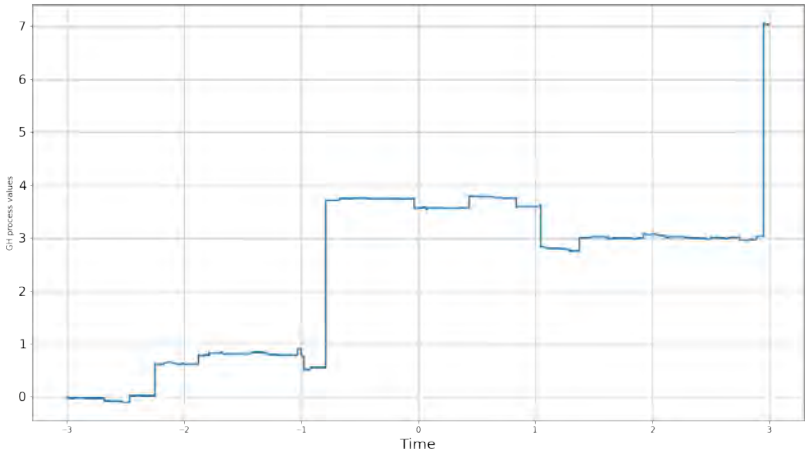


Fig. 4.29 GH process with subordinator from Figure 4.27

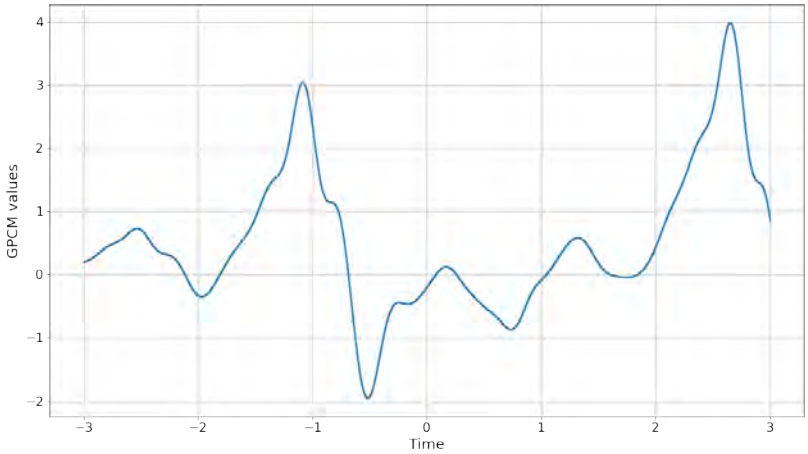


Fig. 4.30 Convolution of the filter in Figure 4.28 and the GH process in Figure4.29.

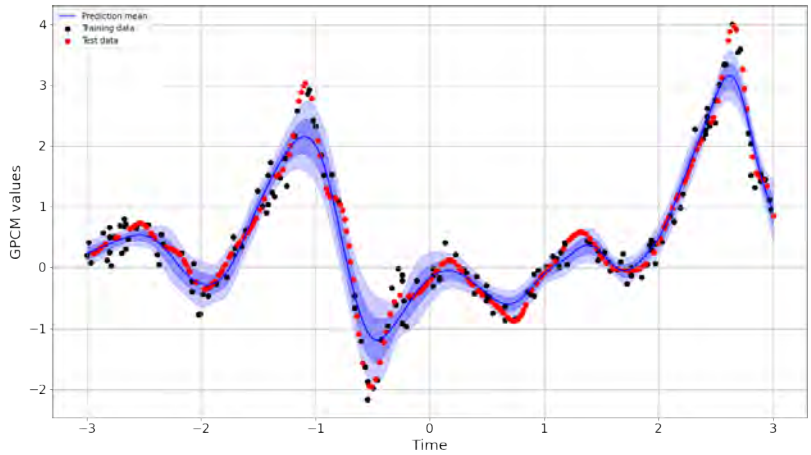


Fig. 4.31 Predictive mean of the GPCM. Error bars are at one and two standard deviations. Training points are in black and test points in red.

predictions for the GPCM in Figure 4.31. Visually, the resulting model's performance is satisfactory. It does not seem to capture the peaks in the data. Comparing the RMSEs of the resulting convolution on the training and test sets we see that on the training set, the RMSE was 0.3067, and on the test set it was 0.2523. In particular, the model does not seem to be overfitting to the training data.

We note that in the presence ot noisy data, the model did not seem to recover the filter or the Lévy processes well. We show the distribution of the predictions for the the filter, Figure 4.32, and paths of the samples for the subordinator process, Figure 4.33, and for the GH process, Figure 4.34. We also show predictions for the spectrum of the gpcm in Figure 4.35 and for the filter 4.36. It is not necessary for the model to recover the filter and GH process in order to perform well, since one is able to make up for deficiencies in the other. As expected, because of the DEQ kernel, our model only contains spectral information at low frequencies. We see that we model the low frequency part of the spectrum quite well for both the GPCM and the filter, but again our mode does not capture higher frequency information. It is possible that in this example the modelling of the spectrum is particularly poor since the filter is not sharply peaked; compare this example with Figure 4.22, where the filter was sharply peaked.

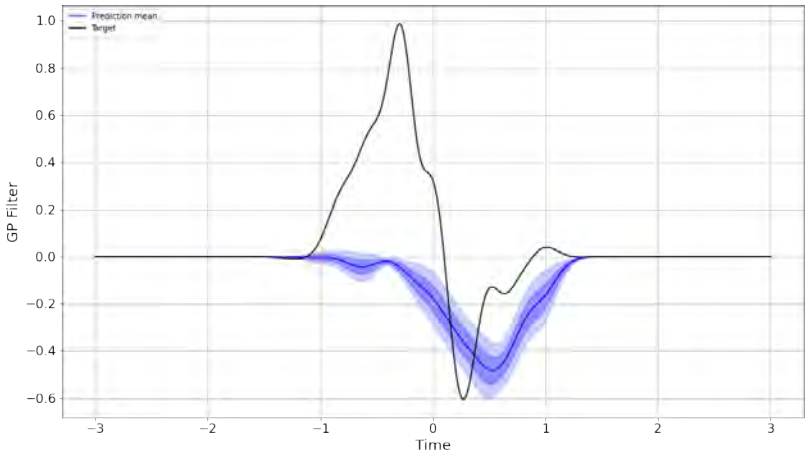


Fig. 4.32 The target filter is shown in black, while the mean of the sampled filters is in blue. Error bars are one and two standard deviations.

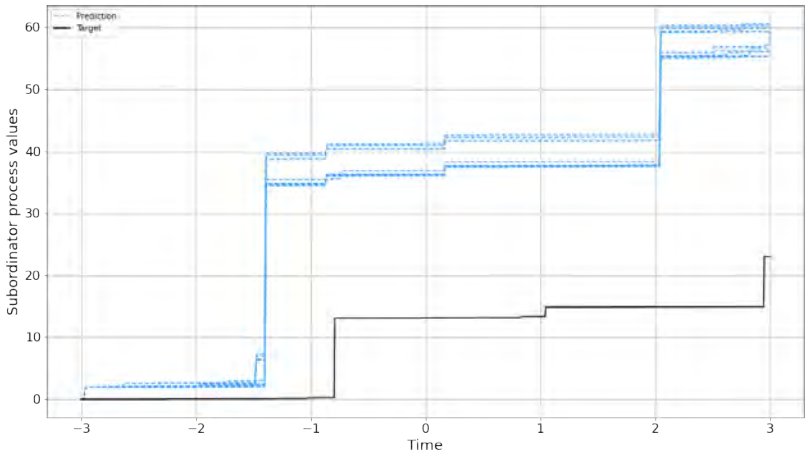


Fig. 4.33 Samples of the GIG process when there is missing and noisy data.

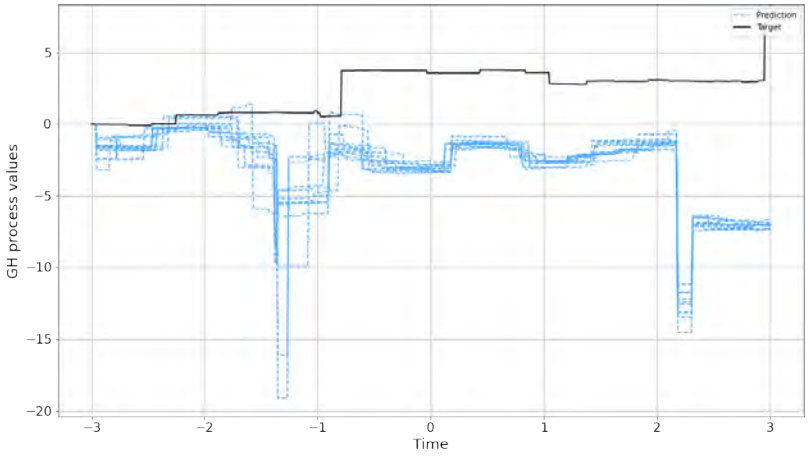


Fig. 4.34 Samples of the GH process.

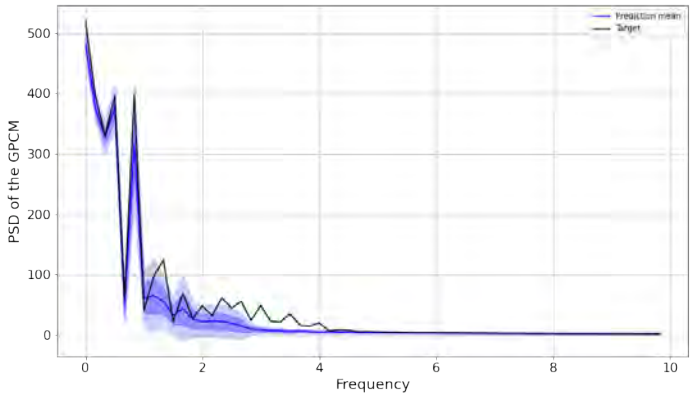


Fig. 4.35 Specturm of the target GPCM in black and the mean of the spectrums of the samples in blue. Error bars are one and two standard deviations.

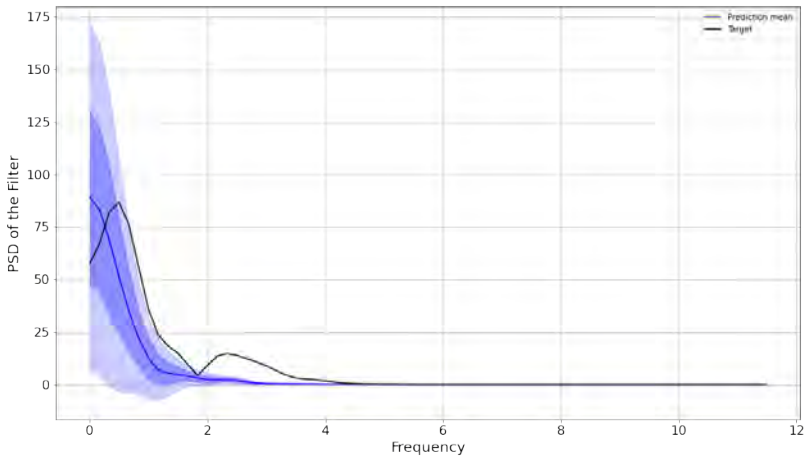


Fig. 4.36 Mean of PSDs of the sampled filters. Error bars at one and two standard deviations.

Chapter 5

Experiments with Real-World Data

5.1 Crude oil prices

In this section we evaluate the ability for the Lévy driven GPCM to predict crude oil prices. The experiment we conduct is similar to one done in Bruinsma et al. (2022); however, we do not train our model on as much data. For a given year (2013), we train a Lv́ey driven GPCM on the oil prices for last half of 2012, the first half of the 2014, and every odd week of the 2013. We withhold the oil prices for the even weeks in 2013 for testing.

Preliminary tests showed that the model learned the data reasonably well, but it gave poor error estimates. This is what motivated attempting to “learn" the parameters by increasing the marginal likelihood. All the result that we show here use this idea. A clear deficiency in our model is that we do not learn the parameters of the Lévy process, and so they need to be set by hand.

We show the result of the experiment with $\lambda = 6$ in Figure 5.1. Judging purely visually, it seems that the curve of the sample mean is fluctuates more at small scales than the standard GPCM which is driven by white noise, but not as much as either the CGPCM or the RGPCM. Examining the average of the spectrum of the samples, we observe that it becomes quite smooth after the low frequencies, partially confirming this observation, Figure 5.2.

To determine whether this behaviour is the same at when λ is greater, and the when the underlying subordinator has heavier tails, we show the GPCM and and some samples of the spectrum for the model when $\lambda = 18$, Figures 5.3 and 5.4. See Table 5.1 for the errors for different values of λ. The heavier tailed model seems to model peaks in the data better than the lighter tailed model with $\lambda = 6$. However, it does not seem to model small flucuations in the data any better. Examining samples of the spectrum in Figure 5.4 we observe that typically the samples become quite smooth very quickly.

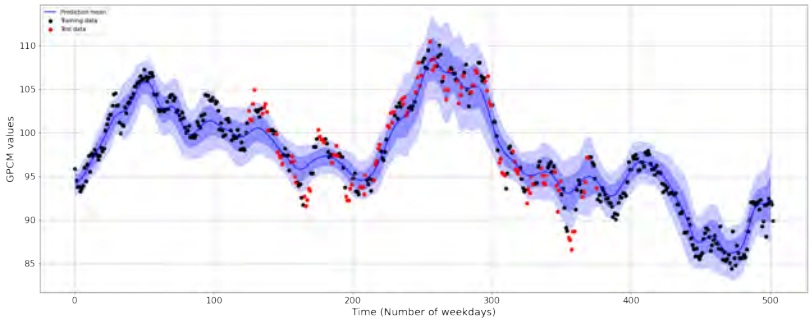


Fig. 5.1 GPCM. Trained assuming that the parameters of the underlying GIG process are λ=6, $\delta = 1$ and $\gamma = 0.1$. Error bars at one and two standard deviations.

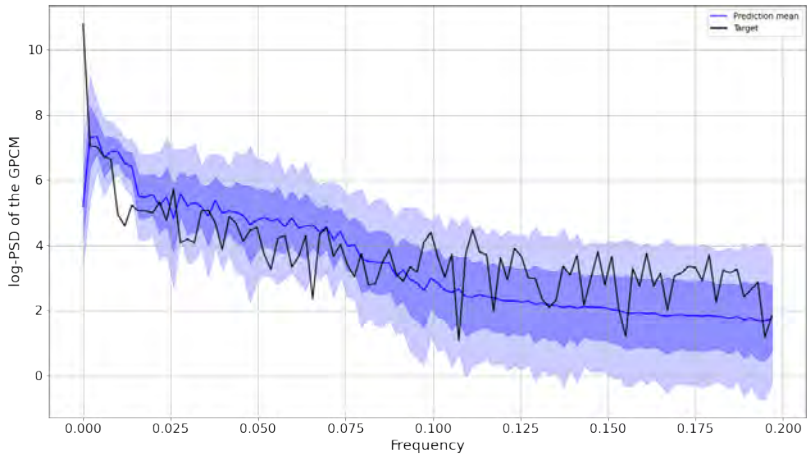


Fig. 5.2 Statistics of the spectrum of the samples for the Lévy driven GPCM with $\lambda = 6, \delta = 1$ and $\gamma = 0.1$ Error bars at one and two standard deviations.

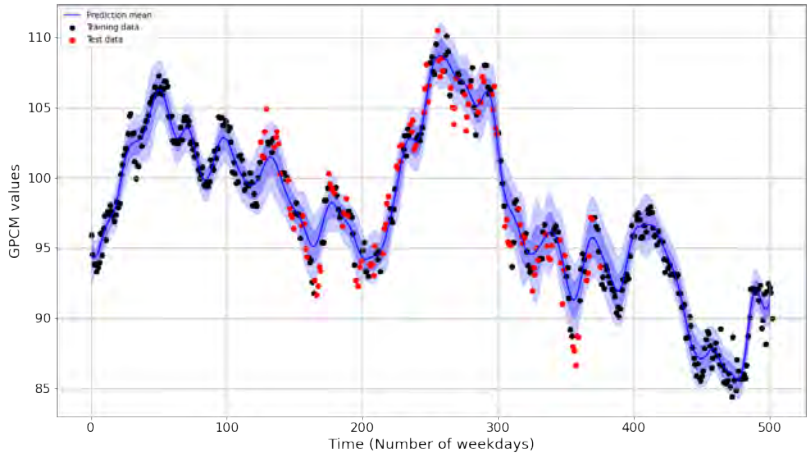


Fig. 5.3 The GPCM of the crude oil data with $\lambda = 18, \delta = 1$ and $\gamma = 0.1$. Error bars at one and two standard deviations.

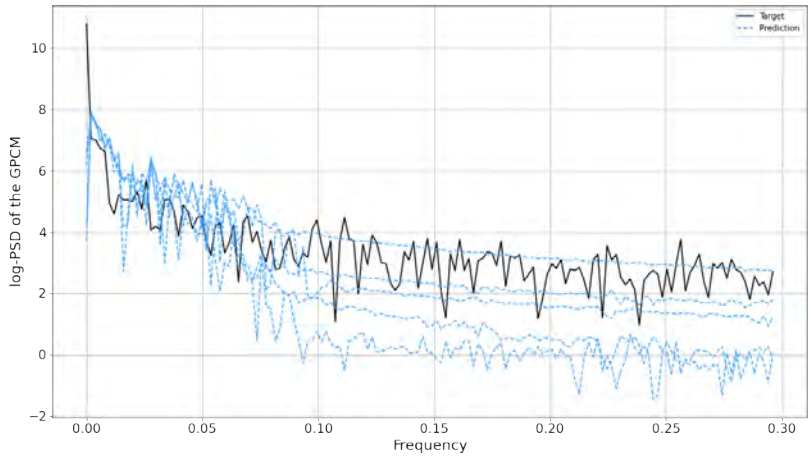


Fig. 5.4 Samples of the spectrum of the GPCM for the crude oil data when $\lambda = 18, \delta = 1$ and $\gamma = 0.1$.

λ	-0.6	6	9	18
RMSE (Test data)	2.24	1.90	1.70	1.70
RMSE (Training data	1.46	1.20	1.13	1.04

Table 5.1 RMSE for 2013 for the Lévy driven GPCMs with λ varying while the other parameters are held fixed at $\delta = 1$ and $\gamma = 0.1$

We see that the samples of the subordinator are not exploring the possible subordinators as well as when we used synthetic data, Figure 5.5. At least the large jumps are fixed. This is somewhat compenstated for by the α_i, Figure 5.6. We show the sample mean with error bars of for the filter in Figure 5.7. The sampler seems to be exploring the space of possible filters well.

We found that as we increased λ that the performance of our model improved somewhat; however, our models did not perform as well as the GPCMs of Tobar et al. (2015) or Bruinsma et al. (2022). This may be because they trained their models on several years of data.

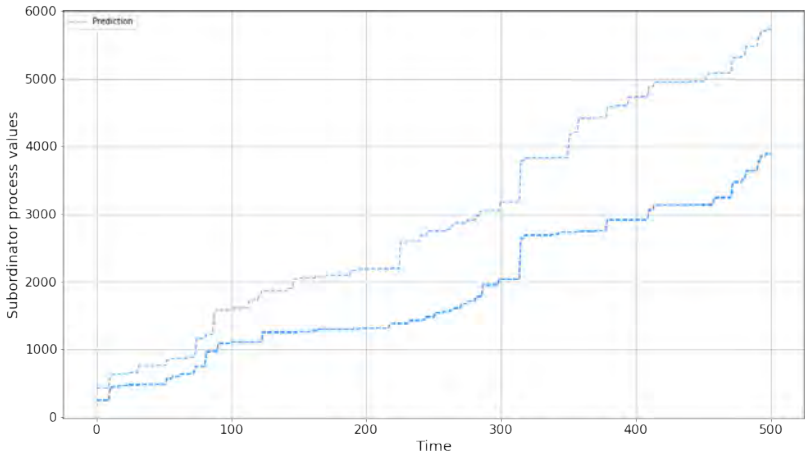


Fig. 5.5 Samples of the subordinator process for the GPCM for the crude oil data when $\lambda = 18, \delta = 1$ and $\gamma = 0.1$.

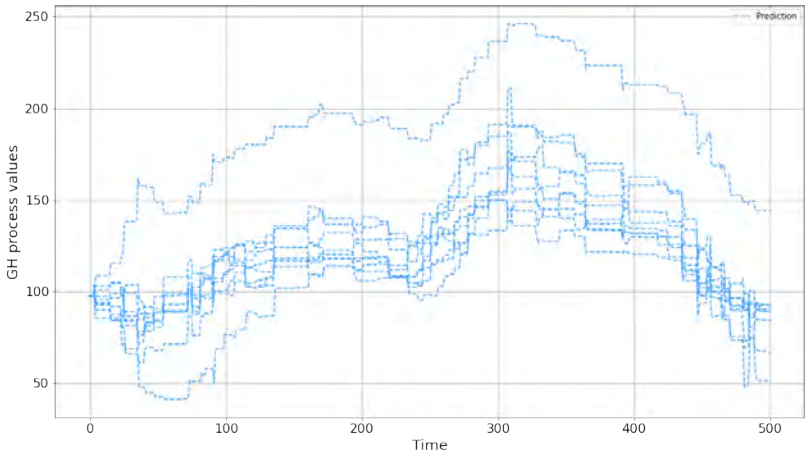


Fig. 5.6 Samples of the GH process for the GPCM for the crude oil data when $\lambda = 18, \delta = 1$ and $\gamma = 0.1$.

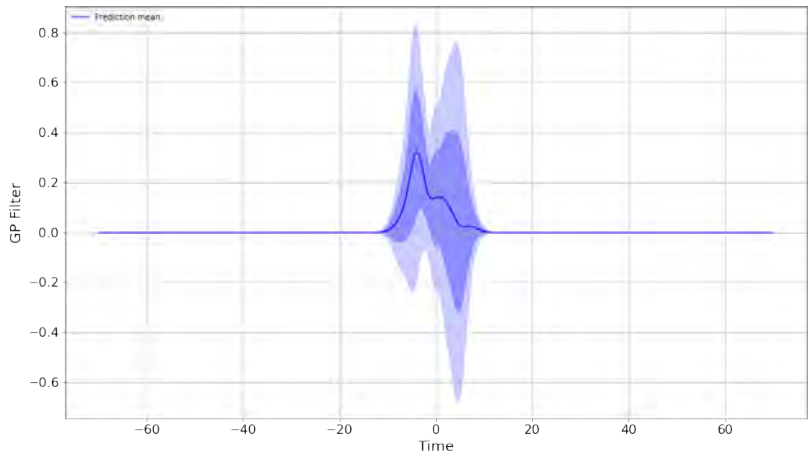


Fig. 5.7 The mean and error bars at one and two standard deviations for the samples of the filter for the GPCM trained on the oil data with $\lambda = 18$, $\delta = 1$ and $\gamma = 0.1$.

Chapter 6

Discussion and Conclusion

In this thesis, we have developed a Gibbs sampling algorthim for Lévy driven GPCMs. These are linear time-invariant systems, whose filter h is a draw from a Gaussian process, and whose input is Lévy noise. We have focussed on inputs which are generalized hyperbolic processes. Essential to this work are the methods of Godsill and Kındap (2021); Kındap and Godsill (2022a) which provide algorithms for sampling paths from GIG processes, and hence from GH processes as well. For simplicity, we focussed on GH processes $\{X(t) : t \geq 0\}$, which are obtained from a normal mean-variance mixture with GIG mixing distribution of a simple form, so that the jumps of X may be expressed as $dX_{\tau_j} = \alpha_j \sqrt{dW\tau_j})$, where dW_{τ_j} is the size of the jump of the subordinator at time τ_j and $\alpha_j \sim \mathcal{N}(0,1)$. We will let α denote the collection of all the α_i, $i = 1,2,3,\ldots$.

This formulation makes it makes it possible for us to develop a Gibbs sampler where we sample W, α and h, successively, from the appropriate marginal distributions. One important aspect of these models is that two of the distributions we need to sample from are Gaussian. We sample α directly from a Gaussian distribution, and sample the values of h at a set of inducting times, and then use GP regression to obtain a sample of h. To generate samples of W we make use of a MH within Gibbs algorithm to sample the jumps of W successively from a partition of the its domain.

In preliminary experiments we found that in the absence of noise on synthetic data generated by a known GPCM, our Gibbs sampler seems to explore the GIG processes, the GH processes and filters well, and it produce varied examples, which seem sensible when compared to the target objects. We also observed that in the absense of noise and when the filter is sharply peaked, that the model does a decent job of recovering the filter. However, when the filter is not sharply peaked, *e.g.* Figure 4.13 where there is no noise, the GPCM does not recover the spectral information in the data beyond the low frequencies.

There are important questions about these models which still need to be addressed and the algorithm we have developed is quite flexible, which opens up pathways to to future work in this direction.

Applications. To what datasets are these models best applied, and what are the limitations of these models? At the current stage of development, this is a difficult question to address as to apply these models effectively, one needs to fine tune parameters by hand. However, it may be that as they stand these models are not particularly effective compared to other available models as they are unable to capture spectral information beyond rather low frequencies.

Inferring the parameters. While we experimented with improving the parameters of the filter by performing gradient ascent on the marginal likelihood with each sweep of the Gibbs sampler, there are methods for inferring parameters using an MCMC approach which would fit in better with our inference scheme, and may perform better. Moreover, we only experimented with learning the parameters of the filter, and since we saw that performance improvements can be gained by considering different parameters of the Lévy process, to be able to exploit these models it will be be necessary to incorporate inference for the parameters of the L'evy processes.

Variants of the model. The GPCM was introduced in Tobar et al. (2015), as the convolution between a Gaussian process, and white noise. We chose to investigate this model, replacing the white noise with a GH Lévy process. However, this model has several components which can be adjusted. In Bruinsma et al. (2022) two new GPCMs were introduced, the Causal GPCM and the Rough GPCM. In the causal version, the output only depends on inputs from the past, naturally, we could also consider a Lévy driven CGPCM. The RGPCM on the other hand is quite a different model. It is the convolution of a sample from a Gaussian process with covariance function defined by windowed white noise, and a sample from a Gaussian process with Matèrn-1/2 kernel. Naturally on could consider replacing the windowed white noise with a windowed Lévy process. But there are other possibilites, a Gaussian processes with Matérn-1/2 kernel is a Ornstein-Uhlenbeck process driven by the Weiner process, and one could instead take it to be an OU process driven by, for example, a GH Lévy process.

Variational methods. Variational methods were used for inference in Tobar et al. (2015) and Bruinsma et al. (2022). It would be advantageous to develop this approach in the context of Lévy driven GPCMs as it may improve performance and provide us with an effective means of learning parameters.

References

David Applebaum. *Lévy Processes and Stochastic Calculus*. Cambridge Studies in Advanced Mathematics. Cambridge University Press, 2 edition, 2009. doi: 10.1017/CBO9780511809781.

Ole E. Barndorff-Nielsen and Christian Halgreen. Infinite divisibility of the hyperbolic and generalized inverse gaussian distributions. *Zeitschrift für Wahrscheinlichkeitstheorie und Verwandte Gebiete*, 38:309–311, 1977.

Richard F. Bass. *Stochastic Processes*. Cambridge Series in Statistical and Probabilistic Mathematics. Cambridge University Press, 2011. doi: 10.1017/CBO9780511997044.

Jean Bertoin. *Lévy Processes*. Cambridge Tracts in Mathematics. Cambridge University Press, 1 edition, 1996.

Wessel P. Bruinsma, Martin Tegnér, and Richard E. Turner. Modelling non-smooth signals with complex spectral structure, 2022. URL https://arxiv.org/abs/2203.06997.

Roberto Calandra, Jan Peters, Carl Edward Rasmussen, and Marc Peter Deisenroth. Manifold gaussian processes for regression. In *2016 International Joint Conference on Neural Networks (IJCNN)*, pages 3338–3345, 2016. doi: 10.1109/IJCNN.2016.7727626.

Ali Cemgil. A technique for painless derivation of kalman filtering recursions. 07 2001.

Rama Cont and Peter Tankov. *Financial Modelling with Jump Processes*. Chapman and Hall/CRC, 1 edition, 2003.

David Duvenaud, James Robert Lloyd, Roger Grosse, Joshua B. Tenenbaum, and Zoubin Ghahramani. Structure discovery in nonparametric regression through compositional kernel search. In *Proceedings of the 30th International Conference on International Conference on Machine Learning - Volume 28*, ICML'13, page III–1166–III–1174. JMLR.org, 2013.

Ernst Eberlein. Application of generalized hyperbolic lévy motions to finance. In Ole E. Barndorff-Nielsen, Sidney I. Resnick, and Thomas Mikosch, editors, *Lévy Processes: Theory and Applications*, pages 319–336, Boston, MA, 2001. Birkhäuser Boston. ISBN 978-1-4612-0197-7. doi: 10.1007/978-1-4612-0197-7_14. URL https://doi.org/10.1007/978-1-4612-0197-7_14.

Ernst Eberlein and E. A. Hammerstein. Generalized hyperbolic and inverse gaussian distributions: Limiting cases and approximation of processes. In R. C. Dalang, M Dozzi, and F Russo, editors, *Seminar on Stochastic Analysis, Random Fields and Applications IV*, pages 221–264, Basel, Switzerland, 2004. Birkhäuser Basel.

Thomas S. Ferguson and Michael J. Klass. A Representation of Independent Increment Processes without Gaussian Components. *The Annals of Mathematical Statistics*, 43(5): 1634 – 1643, 1972. doi: 10.1214/aoms/1177692395. URL https://doi.org/10.1214/aoms/1177692395.

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

Simon Godsill and Yaman Kındap. Point process simulation of generalised inverse gaussian processes and estimation of the jaeger integral. *Statistics and Computing*, 32(1):13, Dec 2021. ISSN 1573-1375. doi: 10.1007/s11222-021-10072-0. URL https://doi.org/10.1007/s11222-021-10072-0.

Roger Grosse, Ruslan Salakhutdinov, William T. Freeman, and Joshua B. Tenenbaum. Exploiting compositionality to explore a large space of model structures. In Nando de Freitas and Kevin P. Murphy, editors, *Proceedings of the Twenty-Eighth Conference on Uncertainty in Artificial Intelligence, Catalina Island, CA, USA, August 14-18, 2012*, pages 306–315. AUAI Press, 2012. URL http://uai.sis.pitt.edu/displayArticleDetails.jsp?mmnu=1&smnu=2&article_id=2293&proceeding_id=28.

E. Grosswald. The student t-distribution of any degree of freedom is infinitely divisible. *Zeitschrift für Wahrscheinlichkeitstheorie und Verwandte Gebiete*, 36(2):103–109, Jun 1976. ISSN 1432-2064. doi: 10.1007/BF00533993. URL https://doi.org/10.1007/BF00533993.

W. K. Hastings. Monte Carlo Sampling Methods using Markov Chains and their Applications. *Biometrika*, 57(1):97–109, April 1970. doi: 10.1093/biomet/57.1.97.

Phillip A Jang, Andrew Loeb, Matthew Davidow, and Andrew G Wilson. Scalable levy process priors for spectral kernel learning. In I. Guyon, U. Von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017. URL https://proceedings.neurips.cc/paper/2017/file/02b1be0d48924c327124732726097157-Paper.pdf.

Richard W. Katz and Barbara G. Brown. Extreme events in a changing climate: Variability is more important than averages. *Climatic Change*, 21(3):289–302, Jul 1992. ISSN 1573-1480. doi: 10.1007/BF00139728. URL https://doi.org/10.1007/BF00139728.

Yaman Kındap and Simon Godsill. Point process simulation of generalised hyperbolic lévy processes. Manuscript, 2022a.

Yaman Kındap and Simon Godsill. Non-gaussian process regression. Manuscript, 2022b.

Andreas E. Kyprianou. Lévy processes and continuous state branching processes: Part 1. URL https://people.bath.ac.uk/ak257/LCSB/part1.pdf.

P. A. W Lewis and G. S. Shedler. Simulation of nonhomogeneous poisson processes by thinning. *Naval Research Logistics Quarterly*, 26(3):403–413, 1979. doi: https://doi.org/10.1002/nav.3800260304. URL https://onlinelibrary.wiley.com/doi/abs/10.1002/nav.3800260304.

Gustavo Malkomes, Charles Schaff, and Roman Garnett. Bayesian optimization for automated model selection. In D. Lee, M. Sugiyama, U. Luxburg, I. Guyon, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 29. Curran Associates, Inc., 2016. URL https://proceedings.neurips.cc/paper/2016/file/3bbfdde8842a5c44a0323518eec97cbe-Paper.pdf.

B. Mandelbrot. New methods in statistical economics. *Journal of Political Economy*, 71(5): 421–440, 1963.

Nicholas Metropolis, Arianna W. Rosenbluth, Marshall N. Rosenbluth, Augusta H. Teller, and Edward Teller. Equation of state calculations by fast computing machines. *The Journal of Chemical Physics*, 21(6):1087–1092, 1953. doi: 10.1063/1.1699114. URL https://doi.org/10.1063/1.1699114.

Kevin P. Murphy. *Probabilistic Machine Learning: Advanced Topics*. MIT Press, 2023. URL probml.ai.

Chrysostomos L. Nikias and Min Shao. *Signal Processing with Alpha-Stable Distributions and Applications*. Wiley-Interscience, USA, 1995. ISBN 047110647X.

Junier B. Oliva, Avinava Dubey, Andrew G. Wilson, Barnabas Poczos, Jeff Schneider, and Eric P. Xing. Bayesian nonparametric kernel-learning. In Arthur Gretton and Christian C. Robert, editors, *Proceedings of the 19th International Conference on Artificial Intelligence and Statistics*, volume 51 of *Proceedings of Machine Learning Research*, pages 1078–1086, Cadiz, Spain, 09–11 May 2016. PMLR. URL https://proceedings.mlr.press/v51/oliva16.html.

Carl Edward Rasmussen and Christopher K. I. Williams. *Gaussian processes for machine learning*. Adaptive computation and machine learning. MIT Press, 2006. ISBN 026218253X.

Jan Rosiński. *Series Representations of Lévy Processes from the Perspective of Point Processes*, pages 401–415. Birkhäuser Boston, Boston, MA, 2001. ISBN 978-1-4612-0197-7. doi: 10.1007/978-1-4612-0197-7_18. URL https://doi.org/10.1007/978-1-4612-0197-7_18.

Ken-iti Sato. *Lévy Processes and infinitely divisible distributions*. Cambridge Studies in Advanced Mathematics. Cambridge University Press, 1 edition, 1999.

Shengyang Sun, Guodong Zhang, Chaoqi Wang, Wenyuan Zeng, Jiaman Li, and Roger Grosse. Differentiable compositional kernel learning for Gaussian processes. In Jennifer Dy and Andreas Krause, editors, *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pages 4828–4837. PMLR, 10–15 Jul 2018. URL https://proceedings.mlr.press/v80/sun18e.html.

Michalis Titsias. Variational learning of inducing variables in sparse gaussian processes. In David van Dyk and Max Welling, editors, *Proceedings of the Twelth International Conference on Artificial Intelligence and Statistics*, volume 5 of *Proceedings of Machine Learning Research*, pages 567–574, Hilton Clearwater Beach Resort, Clearwater Beach, Florida USA, 16–18 Apr 2009. PMLR. URL https://proceedings.mlr.press/v5/titsias09a.html.

Felipe Tobar, Thang D Bui, and Richard E Turner. Learning stationary time series using gaussian processes with nonparametric kernels. In C. Cortes, N. Lawrence, D. Lee, M. Sugiyama, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 28. Curran Associates, Inc., 2015. URL https://proceedings.neurips.cc/paper/2015/file/95e6834d0a3d99e9ea8811855ae9229d-Paper.pdf.

Andrew Wilson and Ryan Adams. Gaussian process kernels for pattern discovery and extrapolation. In Sanjoy Dasgupta and David McAllester, editors, *Proceedings of the 30th International Conference on Machine Learning*, volume 28 of *Proceedings of Machine Learning Research*, pages 1067–1075, Atlanta, Georgia, USA, 17–19 Jun 2013. PMLR. URL https://proceedings.mlr.press/v28/wilson13.html.

Robert L. Wolpert and Katja Ickstadt. *Simulation of Lévy Random Fields*, pages 227–242. Springer New York, New York, NY, 1998. ISBN 978-1-4612-1732-9. doi: 10.1007/978-1-4612-1732-9_12. URL https://doi.org/10.1007/978-1-4612-1732-9_12.