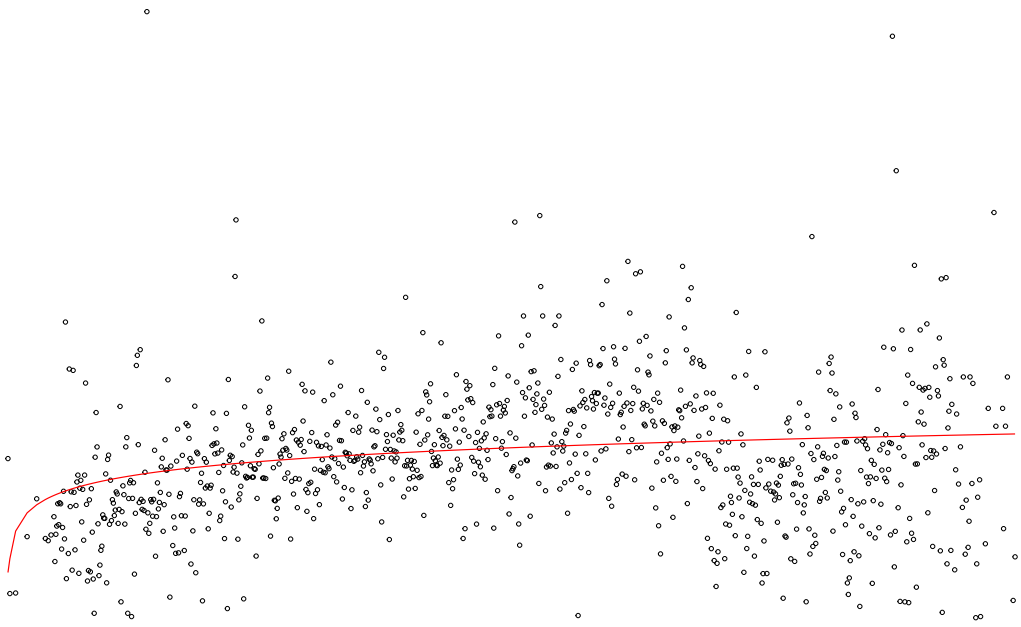
Generalised Random Fields and the De Wijs Process

- Theory and Implementation -

Master Thesis

Nicholas Fitzhugh



Aalborg University
Department of Mathematics

Copyright © Aalborg University 2015

Department of Mathematics
Aalborg University
http://www.aau.dk

AALBORG UNIVERSITY
STUDENT REPORT

Title:
Generalised Random Fields and the De Wijs Process

Theme:
Master Thesis

Project Period:
Spring Semester 2019

Participant:
Nicholas Fitzhugh

Supervisor:
Rasmus Plenge Waagepetersen

Page Numbers: 58

Date of Completion:
June 6, 2019

Abstract:

In this thesis, we present the theory of generalised functions as a foundation of generalised stochastic processes. Afterwards, we introduced the generalised stochastic process, which serves as an abstraction of the conventional notion of stochastic processes. A particular case of generalised stochastic processes is the generalised random field, where a special case of these is of particular interest. Specifically, a so-called conformal model, called the De Wijs plus white noise process, is the centre of attention in the thesis. We present theory on parameter estimation for this process and seek to apply this to a particular dataset. To do this, we implement two different estimation method in the statistical programming language R. We then attempt to utilise these implemented functions on the so-called `bcicov`-dataset, which contains measurements of soil samples, from Barro Colorado Island in Panama.

The content of this report is freely available, but publication (with reference) may only be pursued due to agreement with the author.

Contents

Preface

This project is a Masters Thesis produced by a student on the fourth semester of the Master's Programme in Mathematics on Aalborg University. The thesis is produced during the spring semester in 2019. The project is mainly addressed to students (and others) who already have a basic understanding of the notion of stochastic processes, random fields, as well as statistical programming using the programming language R.

The thesis consists of five main chapters. In Chapter 1 we introduce the notion of generalised functions, as a foundation for the second chapter. In the second chapter, we detail a generalisation of the usual concept of stochastic processes. In addition to this, we introduce the so-called generalised random fields, one of which are of central importance in the thesis. In Chapter 3, we implement the estimation methods detailed in the second chapter, using the programming language R. In Chapter 4, we utilise the implementations on example data, which is introduced in Section 3.1. Lastly, in Chapter 5, we discuss the results and provide closing remarks on the findings of the thesis.

Aalborg University, June 6, 2019

Nicholas Fitzhugh
<nfitzh14@student.aau.dk>

Introduction

In geostatistics, spatial data is often modelled using so-called random fields. Often these kinds of models are sufficient. Sometimes, however, a more general model is necessary. The usual random fields assume that observations stem from a single point, typically either in $\mathbb{R}^2$ or $\mathbb{R}^3$, but data is often collected from an area with non-zero area or volume. Thus it can be useful to step away from the usual application of the random fields and instead, use the more general so-called generalised random fields. Generalised random fields are a special case of a generalisation of the usual notion of stochastic processes.

In the infant years of geostatistics, the work of Georges Matheron gathered much attention and popularity. In Matheron [1962] and Matheron [1971] the notion of the De Wijs process is first pioneered, and was, during this time, very popular. As attention to the De Wijs process died down, an article at the turn of the millennium, McCullagh [2002], sparked interest in the De Wijs process for practical application. During this period of attention, mathematicians such as Debashis Modal and Julian Besag show the connection between so-called Gaussian Markov Random Fields, and the De Wijs process (see, e.g. Besag and Mondal [2005]). On the application front, David Clifford and Peter McCullagh utilised the so-called De Wijs plus white noise process to analyse crop yields in Clifford and McCullagh [2006]. In the later talk, the latter would refer to the process as a *loi du terroir*[1], because of its supposed universality in modelling crop yield independently of crop type or season. The abstract for the talk in question can be seen in McCullagh [2003].

With crop yield data being closely related to nutrients and resources in the soil in which these crops are planted, one might hypothesise that the gold-standard for crop yield data is applicable on data describing the contents of soil measurements.

[1] French for "law of the soil".

Chapter 1

Generalised Functions

In this chapter we introduce so-called *Generalised Functions*. Before these can be defined we must first define so-called *test functions*. The chapter is based on Schäffler [2018, Chapter 1].

1.1 Test Functions

In the following, let $\mathcal{C}_d^\infty$ denote all infinitely differentiable (i.e. smooth) continuous functions from $\mathbb{R}^d$ to $\mathbb{R}$. We define the support of such a function f by

$$\operatorname{supp}(f) = \operatorname{cl}\Big(\{x \in \mathbb{R}^d : f(x) \neq 0\}\Big),$$

where $\operatorname{cl}(\cdot)$ denotes the closure of a set.
We may then define test functions in the following way.

Definition 1.1 (Test Functions). *A function $\varphi \in \mathcal{C}_d^\infty$ is called a test function, if $\operatorname{supp}(\varphi)$ is bounded. The set of test functions from $\mathbb{R}^d$ to $\mathbb{R}$ is denoted $\mathcal{T}_d$.*

Note that $\mathcal{T}_d \subset \mathcal{C}_d^\infty$. Let $\varphi \in \mathcal{T}_d$, and consider the support of the scalar multiple $a\varphi$, which is

$$\operatorname{supp}(a\varphi) = \begin{cases} \operatorname{supp}(\varphi), & a \neq 0 \\ \emptyset, & a = 0. \end{cases}$$

Thus $a\varphi \in \mathcal{T}_d$. Now let $\varphi_1, \varphi_2 \in \mathcal{T}_d$, and consider the addition $\varphi_1 + \varphi_2$. Considering the support of the addition of the test functions, we have that

$$\operatorname{supp}(\varphi_1 + \varphi_2) \subset \operatorname{supp}(\varphi_1) \cup \operatorname{supp}(\varphi_2).$$

Since both $\operatorname{supp}(\varphi_1)$ and $\operatorname{supp}(\varphi_2)$ are bounded, it follows that $\operatorname{supp}(\varphi_1 + \varphi_2)$ is bounded, and thus the addition $\varphi_1 + \varphi_2$ is a test function. This, along with other properties which hold trivially, shows that $\mathcal{T}_d$ is a vector space.
We now introduce an important definition regarding the meaning of convergence of a sequence of test functions.

Definition 1.2 (Convergence of Test Functions). *Let $\{\varphi_i\}_{i\in\mathbb{N}}$ be a sequence of test functions. The sequence is said to be convergent to a function $\varphi \in \mathcal{T}_d$, if*

- *There exists a bounded set $M \subset \mathbb{R}^d$, such that $\operatorname{supp}(\varphi_i) \subset M$ for all $i \in \mathbb{N}$.*

- *The sequence $\{\varphi_i - \varphi\}$ converges uniformly to the zero function in $\mathbb{R}^d$.*
- *All partial derivatives of any fixed order of $(\varphi_i - \varphi)$ converges uniformly to the zero function in $\mathbb{R}^d$.*

It turns out that it is possible to approximate any continuous function, with compact support, with a sequence of test functions. To prove this, we first introduce the function $\psi : \mathbb{R}^d \to \mathbb{R}$ defined by,

$$\psi(x) = \begin{cases} 0, & \text{when } 1 \leq \|x\|^2, \\ \exp(-\frac{1}{1-\|x\|^2}), & \text{when } 1 > \|x\|^2, \end{cases}$$

where $\|\cdot\|$ is the Euclidean norm. Note that ψ is symmetric, and has bounded support. As it is also smooth, $\psi \in \mathcal{T}_d$. Let $I_\psi = \int_{\mathbb{R}^d} \psi(x)\mathrm{d}x < \infty$. Furthermore, define

$$\psi_1(x) = \frac{\psi(x)}{I_\psi} \text{ for } x \in \mathbb{R}^d.$$

We use these functions to define a function which is central for approximation of continuous functions using test functions,

$$\psi_R(x) = \frac{\psi_1(x/R)}{R^d} \tag{1.1}$$

with $R > 0$. The support of ψ_R is the d-dimensional ball $\bar{B}(0, R)$, thus it is also a test function. Considering the integral of $\psi_R(x)$, using integration by substitution and setting $y = \frac{x}{R}$, we have

$$\int_{\mathbb{R}^d} \psi_R(x)\mathrm{d}x = \int_{\mathbb{R}^d} \frac{\psi_1(\frac{x}{R})}{R^n}\mathrm{d}x = \int_{\mathbb{R}^d} \frac{\psi_1(y)}{R^n} \det\left(J(y)\right) \mathrm{d}y = 1,$$

since $J(y)$ is the Jacobian matrix of the function $x = yR$, which is just a matrix with R on the diagonal.
We can now formulate a theorem about the approximation of continuous functions, with compact support, using test functions.

Theorem 1.3 (The approximation theorem). *Let $f : \mathbb{R}^d \to \mathbb{R}$ be a continuous function with compact support. For each $\varepsilon > 0$, there exists a function $\varphi \in \mathcal{T}_d$, such that,*

$$|f(x) - \varphi(x)| < \varepsilon, \text{ for all } x \in \mathbb{R}^d.$$

Specifically, defining

$$\varphi_R(x) = \int_{\mathbb{R}^d} f(u)\psi_R(u - x)\mathrm{d}u,$$

it holds that,

$$\lim_{R \to 0} \varphi_R(x) = f(x), \text{ for all } x \in \mathbb{R}^d,$$

uniformly.

Proof. The following proof is a more detailed version of the proof for [Schäffler, 2018, Thm. 2.1].

As f is continuous with compact support, it is also uniformly continuous(see e.g [Poulsen, 2015, Thm. 6.29]), meaning that for each $\varepsilon > 0$ there exists a $\delta > 0$ such that

$$|f(x) - f(y)| < \varepsilon, \text{ for all } x, y \in \mathbb{R}^d \text{ with } \|x - y\| < \delta.$$

Additionally, we have that

$$\begin{aligned}\varphi_R(x) &= \int_{\mathbb{R}^d} f(u)\psi_R(u-x)\mathrm{d}u = \int_{\bar{B}(x,R)} f(u)\psi_R(u-x)\mathrm{d}u \\ &= f(y)\int_{\bar{B}(x,R)} \psi_R(u-x)\mathrm{d}u = f(y) \text{ for some } y \in \bar{B}(x,R).\end{aligned}$$

where the second equality is due to the support of $\psi_R(u-x)$ being $\bar{B}(x,R)$, and the third equality is from the mean value theorem of integrals (see e.g Amann and Escher [2008, Ch. 6.4]). The last equality is due to (1.1).
With $R < \delta$, we have

$$|f(x) - \varphi_R| = |f(x) - f(y)| < \varepsilon, \text{ because } \|x - y\| \leq R < \delta.$$

Thus as $R \to 0$ the function $\varphi_R \to f$ uniformly, proving the result. □

For an example of this, consider the function

$$f(x) = \begin{cases} 1 - \frac{1}{2}|x|, & \text{when } |x| < 2, \\ 0, & \text{when } |x| \geq 2. \end{cases} \tag{1.2}$$

Since $f(x)$ is continuous with compact support, we may approximate it using φ_R as defined in Theorem 1.3. The plot of $f(x)$ can be seen in Figure 1.1. In the plots in Figure 1.2 and Figure 1.3, we can see that φ_R approximates $f(x)$ better for the lower value of R. Note that the integral that defines φ_R has been approximated numerically, resulting in the rather jagged shape in Figure 1.2. For large values of R the function φ_R is a smoothed version of $f(x)$.

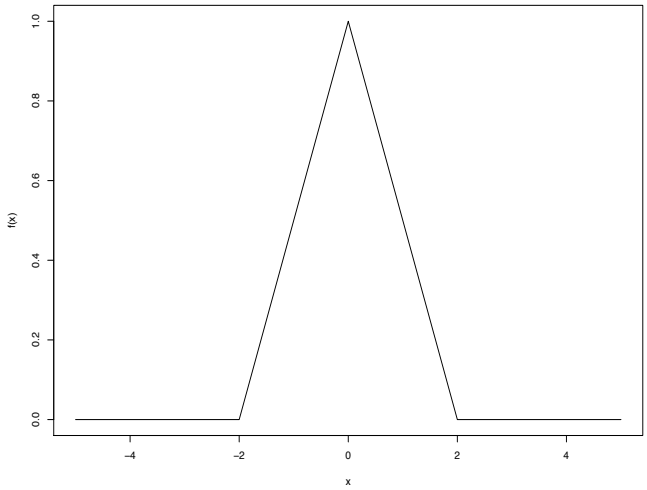


Figure 1.1: Plot of the function $f(x)$ as defined in (1.2)

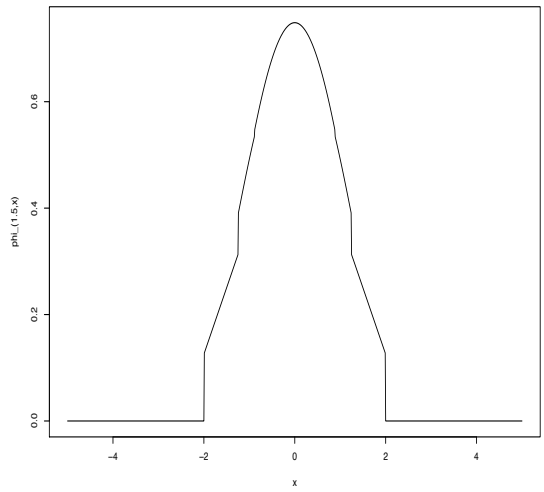


Figure 1.2: Plot of the function $\varphi_{1.5}(x)$.

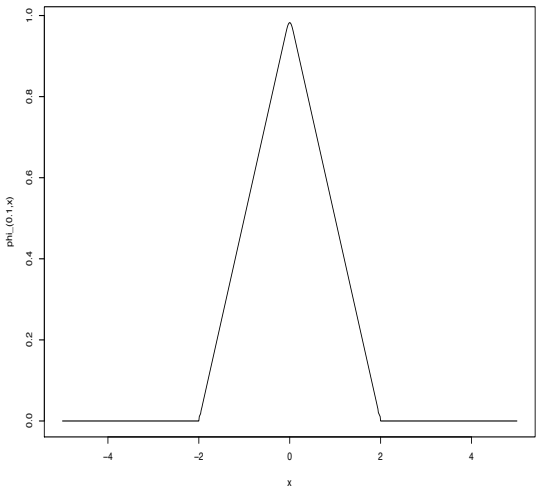


Figure 1.3: Plot of the function $\varphi_{0.1}(x)$.

Test functions are of central importance in the next section, where so-called functionals are introduced.

1.2 Functionals

In this section, we introduce the notion of functionals. We define the term in a very general setting. Before defining functionals, we note that $\mathcal{A}$ being a vector space over $\mathcal{B}$, means that, among other things, $\mathcal{A}$ is closed under multiplication with elements of $\mathcal{B}$.

Definition 1.4 (Functional). *For a given vector space $\mathcal{A}$ over $\mathcal{B}$, a mapping*

$$F : \mathcal{A} \to \mathcal{B},$$

is called a functional.

Furthermore, a functional F is said to be linear if for all $a \in \mathcal{B}$, and $x, y \in \mathcal{A}$,

$$F(ax) = aF(x)$$
$$F(x+y) = F(x) + F(y).$$

With functionals defined in a general setting, we turn our attention to a specific functional. For this purpose, let $f : \mathbb{R}^d \to \mathbb{R}$ be a locally integrable function, meaning that $|f|$ is integrable on the closed ball, $\bar{B}(x, \varepsilon)$ with centre x and radius ε, for each $x \in \mathbb{R}^d$ and each $\varepsilon > 0$.

For each such f, we may define the mapping $F_f : \mathcal{T}_d \to \mathbb{R}$, by

$$F_f(\varphi) = \int_{\mathbb{R}^d} f(x)\varphi(x)\mathrm{d}x \tag{1.3}$$

Since $\mathcal{T}_d$ is a vector space over $\mathbb{R}$, as shown in Section 1.1, the mapping F_f defines a functional. Furthermore, F_f is a linear functional, since for $a \in \mathbb{R}$ and $\varphi_1, \varphi_2 \in \mathcal{T}_d$ we have

$$F_f(a\varphi) = aF_f(\varphi)$$
$$F_f(\varphi_1 + \varphi_2) = \int_{\mathbb{R}^d} f(x)\left(\varphi_1(x) + \varphi_2(x)\right)\mathrm{d}x = F_f(\varphi_1) + F_f(\varphi_2).$$

A functional defined as F_f can be used to give meaning to the derivative of a function which is not differentiable. We show this in the case where $d = 1$ in the following way. Suppose $f : \mathbb{R} \to \mathbb{R}$ is locally integrable as before, and also differentiable. Suppose furthermore, that the derivative f' is locally integrable, then, using integration by parts, we have

$$\int_{\mathbb{R}} f'(x)\varphi(x)\mathrm{d}x = [f(x)\varphi(x)]_{-\infty}^{\infty} - \int_{\mathbb{R}} f(x)\varphi'(x)\mathrm{d}x. \tag{1.4}$$

Since $\varphi \in \mathcal{T}$, the set $\mathrm{supp}(\varphi)$ is bounded. Thus the first term in (1.4) is zero, and

$$\int_{\mathbb{R}} f'(x)\varphi(x)\mathrm{d}x = -\int_{\mathbb{R}} f(x)\varphi'(x)\mathrm{d}x,$$

meaning $F_{f'}(\varphi) = -F_f(\varphi')$. In the above, the left hand side is defined for differentiable functions f, but the right hand side is defined for any locally integrable functions f.

Now let $\varphi(u) = \psi_{R,x}(u)$, where $\psi_{R,x}(u)$ is defined as

$$\psi_{R,x}(u) = \psi_R(u - x), \tag{1.5}$$

where ψ_R is defined as in (1.1).
For a differentiable function f, where f' is locally integrable, we have

$$-F_f(\psi'_{R,x}) = F_{f'}(\psi_{R,x}) = \int_{\mathbb{R}} f'(u)\psi_{R,x}(u)\mathrm{d}u = \int_{\bar{B}(x,R)} f'(u)\psi_{R,x}(u)\mathrm{d}u = f'(y)\int_{\bar{B}(x,R)} \psi_{R,x}(u)\mathrm{d}u$$
$$= f'(y)$$

for some $y \in \bar{B}(x, R)$, where the fourth equality is a result of the mean value theorem for integrals. Taking the limit as $R \to 0$, we have that,

$$\lim_{R\to 0} -F_f(\psi'_{R,x}) = f'(x), \text{ for } x \in \mathbb{R},$$

which can be used to approximate the derivative of non-differentiable functions.

We now introduce the notion of continuity of functionals from the set of test functions to the real numbers.

Definition 1.5 (Continuity of Functionals). *A functional $F : \mathcal{T}_d \to \mathbb{R}$ is said to be continuous if*

$$\lim_{i\to\infty} F(\varphi_i) = F(\varphi),$$

for all sequences $\{\varphi_i\}$ which are convergent to a function $\varphi \in \mathcal{T}_d$ (see Definition 1.2).

Using the Definition 1.5 it can be shown that functionals on the form in (1.3) are continuous.

Proposition 1.6. *Let $f : \mathbb{R}^d \to \mathbb{R}$ be a locally integrable function. A linear functional $F_f : \mathcal{T}_d \to \mathbb{R}$ on the form,*

$$F_f(\varphi) = \int_{\mathbb{R}^d} f(x)\varphi(x)\mathrm{d}x,$$

is continuous.

Proof. This proof is a collection of arguments from [Schäffler, 2018, p.13], where further details have been added.
The zero function, $\varphi(x) = 0$ for all $x \in \mathbb{R}^d$ is a test function, as

$$\operatorname{supp}(\varphi) = \emptyset,$$

which is closed and bounded. Thus the functional F is continuous if,

$$\lim_{i\to\infty} F(\varphi_i) = 0,$$

for all sequence of test functions which converge to the zero function (see Definition 1.2).

Now let $\{\varphi_i\}_{i\in\mathbb{N}}$ be some sequence of test functions that converges to the zero function. Then for some $\varepsilon > 0$

$$\operatorname{supp}(\varphi_i) \subseteq B(0,\varepsilon), \text{ for all } i \in N.$$

Using this fact, we have that

$$\begin{aligned}\lim_{i\to\infty} |F_f(\varphi_i)| &= \lim_{i\to\infty} \left| \int_{\mathbb{R}^d} f(x)\varphi_i(x)\mathrm{d}x \right| = \lim_{i\to\infty} \left| \int_{B(0,\varepsilon)} f(x)\varphi_i(x)\mathrm{d}x \right| \\ &\leq \lim_{i\to\infty} \int_{B(0,\varepsilon)} |f(x)||\varphi_i(x)|dx \leq \lim_{i\to\infty} \left(\sup_{x\in B(0,\varepsilon)} |\varphi_i(x)| \int_{B(0,\varepsilon)} |f(x)|\mathrm{d}x \right) = 0,\end{aligned}$$

where the last inequality results from test functions being bounded, and the last equality from $|\varphi_i| \to 0$ as $i \to \infty$, proving the result.

□

Next, we see how functionals lead to the definition of so-called generalised functions.

1.3 Generalised Functions

As discussed in the previous section we can define a continuous and linear functional for a locally integrable function f, from the space of test functions to the real numbers. This particular functional is a so-called generalised function, which we now formally define for the space of test functions.

Definition 1.7 (Generalised Functions from $\mathcal{T}_d$). *A linear and continuous functional $F : \mathcal{T}_d \to \mathbb{R}^d$ is called a generalised function. Furthermore, if there exists a locally integrable function, f, such that $F = F_f$, then F is called a regular generalised function.*

The set of generalised function from $\mathcal{T}_d$ is denoted $\mathcal{G}_d$.
Defining the scalar multiple, and addition of generalised function as

$$aF(\varphi) = F(a\varphi), \text{ for } a \in \mathbb{R}, \varphi \in \mathcal{T}_d, F \in \mathcal{G}_d$$
$$(F+G)(\varphi) = F(\varphi) + G(\varphi), \text{ for } \varphi \in \mathcal{T}_d, F, G \in \mathcal{G}_d,$$

the space $\mathcal{G}_d$ is a vector space over $\mathbb{R}$, as all the other requirements are trivially satisfied.
The so-called Dirac distribution, δ_{x_0}[1], is an example of a generalised function. The Dirac distribution is defined as

$$\delta_{x_0}(\varphi) = \varphi(x_0), \text{ for } \varphi \in \mathcal{T}_d \text{ and } x_0 \in \mathbb{R}^d.$$

This generalised function is an example of a generalised function, which is not a regular generalised function. We formulate this minor result as a proposition.

Proposition 1.8. *The Dirac distribution, $\delta_{x_0}(\varphi) = \varphi(x_0)$, is not a regular generalised function.*

Proof. This proof is a collection of arguments from [Schäffler, 2018, p.16], where more detail has been added.
We prove the proposition by contradiction. Therefore assume that there exists a locally integrable function, f, such that

$$\int_{\mathbb{R}^d} f(x)\varphi(x)\mathrm{d}x = \varphi(x_0), \text{ for all } \varphi \in \mathcal{T}_d, \tag{1.6}$$

i.e we assume that the Dirac distribution is a regular generalised function. Since f is locally integrable, then for all $x \in \mathbb{R}^d$ and all $\varepsilon > 0$

$$\int_{\bar{B}(x,\varepsilon)} |f(u)|\mathrm{d}u = d_x < \infty.$$

Thus there exists a small ε such that $d_x < 1$. Consider now the function ψ_{ε,x_0} defined as in (1.5), which is a test function for all $\varepsilon > 0$. Thus by the assumption in (1.6), we must have,

$$\int_{\mathbb{R}^d} f(x)\psi_{\varepsilon,x_0}(x)\mathrm{d}x = \psi_{\varepsilon,x_0}(x_0).$$

[1] Some authors prefer the terminology 'distributions' to refer to generalised functions, however there is no relation to cumulative distribution functions.

However,

$$\int_{\mathbb{R}^d} f(x)\psi_{\varepsilon,x_0}(x)\mathrm{d}x \leq \left|\int_{\mathbb{R}^d} f(x)\psi_{\varepsilon,x_0}(x)\mathrm{d}x\right| \leq \int_{\mathbb{R}^d} |f(x)\psi_{\varepsilon,x_0}(x)|\mathrm{d}x$$
$$\leq \sup_{\bar{B}(x_0,\varepsilon)}|\psi_{\varepsilon,x_0}(x)| \int_{\bar{B}(x_0,\varepsilon)} |f(x)|\mathrm{d}x = \psi_{\varepsilon,x_0}(x_0)d_{x_0} < \psi_{\varepsilon,x_0}(x_0),$$

where third inequality is a result of $|f(x)\psi_{\varepsilon,x_0}(x)| \leq \sup_{\bar{B}(x_0,\varepsilon)}|\psi_{\varepsilon,x_0}(x)||f(x)|$ for all $x \in \mathbb{R}^d$. This proves the result.

□

Chapter 2

Generalised Stochastic Processes

In this chapter, we seek to define the De Wijs process, and therefore also the concept of so-called *generalised random fields*, which is a special case of so-called *generalised stochastic processes*.

2.1 Generalised Stochastic Processes

This section is based on [Schäffler, 2018, Chapter 2.1]. In this section, we define a stochastic process in a very general setting.

Definition 2.1 (Stochastic Processes). *Let $(\Omega, \mathcal{F}, P)$ be a probability space, let $(\Gamma, \mathcal{G})$ be a measurable space and let I be some non-empty set. Then*

$$Z : \Omega \times I \to \Gamma$$

is called a stochastic process with index-set I, if $Z(\cdot, i)$ is a $\mathcal{F}-\mathcal{G}$ measurable function for any fixed $i \in I$.

For a fixed $\omega \in \Omega$ we call the mapping $i \mapsto Z(\omega, i)$ the *path* of Z, and if this function is continuous for all $\omega \in \Omega$ we say that Z has continuous paths (or almost surely continuous paths if it only holds for almost all $\omega \in \Omega$). We usually suppress the dependence on ω and denote $Z(\omega, i)$ simply as $Z(i)$.

Note that the notion of stochastic processes in Definition 2.1 is a very general one. Often the term stochastic process is used to refer only to the case where the index-set is a subset of $\mathbb{R}$. A stochastic process where the index-set is $D \subseteq \mathbb{R}^d$ is called a random field. Thus a stochastic process is usually thought of as a special case of a random field, where $d = 1$, but with Definition 2.1 a random field is a special case of stochastic processes.
Before proceeding we introduce the notion of a *modification*.

Definition 2.2 (Modification). *Let Z and Y be stochastic processes with index-set I. Y is said to be a modification of Z if*

$$P\Big(Z(i) = Y(i)\Big) = 1, \text{ for all } i \in I.$$

When Y is a modification of Z, then obviously Z is also a modification of Y.

As we see in Definition 2.1 we can choose the index set of a stochastic process to be any non-empty set. Specifically, we may choose $I = \mathcal{T}_d$, to obtain random variables indexed over test functions. This is used to define the notion of generalised stochastic processes, but first, we need to define the notion of continuity for the test-function-indexed stochastic process.

Definition 2.3 (Continuity for Test-function-indexed Stochastic Process). *Let Z be a stochastic process with index set $\mathcal{T}_d$. Then Z is said to be continuous if*

$$\lim_{k\to\infty} \mathbb{E}\left[g\left(Z(\varphi_{1k}),\dots,Z(\varphi_{mk})\right)\right] = \mathbb{E}\left[g\left(Z(\varphi_1),\dots,Z(\varphi_m)\right)\right],$$

for any bounded function $g : \mathbb{R}^m \to \mathbb{R}$ and for any m sequences of test-functions such that $\varphi_{ik} \to \varphi_i$ for $k \to \infty$ and $i = 1,\dots,m$ (by $\varphi_{ik} \to \varphi_i$ we mean convergence in the sense of Definition 1.5).

This now enables us to define generalised stochastic processes

Definition 2.4 (Generalised Stochastic Processes). *Let Z be a stochastic process with index-set $\mathcal{T}_d$. If Z is continuous and*

$$Z(a\varphi + b\psi) = aZ(\varphi) + bZ(\psi),$$

almost surely for all $a, b \in \mathbb{R}$ and $\varphi, \psi \in \mathcal{T}_d$, then Z is said to be a generalised stochastic process.

It is clear that from Definition 2.1 generalised stochastic process are *not* generalisations of stochastic processes. Rather they get their name for different reasons. For a fixed $\omega \in \Omega$ we have that a generalised stochastic process $Z(\omega,\cdot)$ is a functional, and thus if Z has continuous paths, $Z(\omega,\cdot)$ is a generalised function for any $\omega \in \Omega$. In fact it turns out, that for any generalised stochastic process, there exists a modification which has continuous paths (in the sense of Definition 1.5) [Dawson, 1970, Theorem 2.1]. We define the mean- and covariance-functional as we would have for a random field,

$$m(\varphi) = \mathbb{E}\left[Z(\varphi)\right], \quad C(\varphi,\psi) = \mathbb{C}\mathrm{ov}\left[Z(\varphi), Z(\psi)\right].$$

Furthermore we note that, as usual, the covariance-functional is symmetric, bi-linear and positive semi-definite, meaning that

$$\sum_{i=1}^{m}\sum_{j=1}^{m} a_i a_j C(\varphi_i,\varphi_j) \geq 0,$$

for all $a_1,\dots,a_m \in \mathbb{R}^m$ and $\varphi_1,\dots,\varphi_m \in \mathcal{T}_d$.
Now let $\varphi_1,\dots,\varphi_n \in \mathcal{T}_d$, and let $F_{1,\dots,n}$ be the distribution function of $(Z(\varphi_1),\dots,Z(\varphi_n))$. Then $F_{1,\dots,n}$ is said to be a finite-dimensional distribution of the generalised stochastic process, Z. If all finite dimensional distributions of Z are Gaussian, then Z is said to be a *Gaussian generalised stochastic process*. With this we can show the following theorem.

Theorem 2.5. *Let $m : \mathcal{T}_d \to \mathbb{R}$ be a continuous linear functional, and let $C : \mathcal{T}_d^2 \to \mathbb{R}$ be a symmetric bi-linear and positive semi-definite functional. Then there exists a Gaussian generalised stochastic process, Z, which satisfies*

$$\mathbb{E}\left[Z(\varphi)\right] = m(\varphi), \text{ for all } \varphi \in \mathcal{T}_d,$$

and

$$\mathbb{C}\text{ov}\left[Z(\varphi), Z(\psi)\right] = C(\varphi, \psi), \text{ for all } \varphi, \psi \in \mathcal{T}_d.$$

Proof. Suppose we have $\varphi_1, \ldots, \varphi_n \in \mathcal{T}_d$. In order to prove the results, we need to show the following:

(i) The distribution is a valid Gaussian distribution, i.e

$$\Sigma = \begin{bmatrix} C(\varphi_1, \varphi_1) & \cdots & C(\varphi_1, \varphi_n) \\ \vdots & \ddots & \vdots \\ C(\varphi_n, \varphi_1) & \cdots & C(\varphi_n, \varphi_n) \end{bmatrix}$$

is positive semidefinite for any choice of $\varphi_1, \ldots, \varphi_n$.

(ii) There exists a stochastic process, Z, with index-set $\mathcal{T}_d$ such that

$$\begin{bmatrix} Z(\varphi_1) \\ \vdots \\ Z(\varphi_n) \end{bmatrix} \sim N\left(\begin{bmatrix} m(\varphi_1) \\ \vdots \\ m(\varphi_n) \end{bmatrix}, \begin{bmatrix} C(\varphi_1, \varphi_1) & \cdots & C(\varphi_1, \varphi_n) \\ \vdots & \ddots & \vdots \\ C(\varphi_n, \varphi_1) & \cdots & C(\varphi_n, \varphi_n) \end{bmatrix} \right),$$

for any $\varphi_1, \ldots, \varphi_n \in \mathcal{T}_d$.

(iii) $Z(a\varphi + b\psi) = aZ(\varphi) + bZ(\psi)$ almost surely for all $\varphi, \psi \in \mathcal{T}_d$ and $a, b \in \mathbb{R}$.

(iv) Z is continuous.

Define a random vector $X : \Omega \to \mathbb{R}^n$ such that

$$X \sim N\left(m, \Sigma\right),$$

where

$$m = \begin{bmatrix} m(\varphi_1) \\ \vdots \\ m(\varphi_n) \end{bmatrix}, \quad \Sigma = \begin{bmatrix} C(\varphi_1, \varphi_1) & \cdots & C(\varphi_1, \varphi_n) \\ \vdots & \ddots & \vdots \\ C(\varphi_n, \varphi_1) & \cdots & C(\varphi_n, \varphi_n) \end{bmatrix}.$$

The matrix is positive semidefinite, since for arbitrary functions $\varphi_1, \ldots, \varphi_n \in \mathcal{T}_d$ we get that for $a, \in \mathbb{R}^n$

$$a^\top \Sigma a = \sum_{i=1}^{n} \sum_{j=1}^{n} a_i a_j C(\varphi_i, \varphi_j) \geq 0,$$

and thus Σ is a positive semi-definite matrix, which proves (i). Furthermore we get that (ii) holds by Kolmogorov's existence theorem. To prove (iii) choose arbitrary $\varphi, \psi \in \mathcal{T}_d$ and $a, b \in \mathbb{R}$. To prove that $Z(a\varphi + b\psi) = aZ(\varphi) + bZ(\psi)$ almost surely we consider the random variable $Z(a\varphi + b\psi) - \left(aZ(\varphi) + bZ(\psi)\right)$. First we note that

$$\begin{aligned} &\mathbb{V}\text{ar}\left[Z(a\varphi + b\psi) - \left(aZ(\varphi) + bZ(\psi)\right)\right] \\ &= \mathbb{V}\text{ar}\left[Z(a\varphi + b\psi)\right] + \mathbb{V}\text{ar}\left[aZ(\varphi) + bZ(\psi)\right] - 2\mathbb{C}\text{ov}\left[Z(a\varphi + b\psi), aZ(\varphi) + bZ(\psi)\right]. \end{aligned} \tag{2.1}$$

We now express each term in (2.1) in terms of the covariance function of Z. By construction we get

$$\mathbb{V}\mathrm{ar}\left[Z(a\varphi + b\psi)\right] = C(a\varphi + b\psi, a\varphi + b\psi). \tag{2.2}$$

The second term can be expanded as follows

$$\begin{aligned}\mathbb{V}\mathrm{ar}\left[aZ(\varphi) + bZ(\psi)\right] &= a^2\mathbb{V}\mathrm{ar}\left[Z(\varphi)\right] + b^2\mathbb{V}\mathrm{ar}\left[Z(\psi)\right] + 2ab\mathbb{C}\mathrm{ov}\left[Z(\varphi), Z(\psi)\right] \\ &= a^2C(\varphi,\varphi) + b^2C(\psi,\psi) + 2abC(\varphi,\psi) \\ &= C(a\varphi + b\psi, a\varphi + b\psi).\end{aligned} \tag{2.3}$$

Finally, for the third term we get

$$\begin{aligned}\mathbb{C}\mathrm{ov}\left[Z(a\varphi + b\psi), aZ(\varphi) + bZ(\psi)\right] &= a\mathbb{C}\mathrm{ov}\left[Z(a\varphi + b\psi), Z(\varphi)\right] + b\mathbb{C}\mathrm{ov}\left[Z(a\varphi + b\psi), Z(\psi)\right] \\ &= aC(a\varphi + b\psi, \varphi) + bC(a\varphi + b\psi, \psi) \\ &= C(a\varphi + b\psi, a\varphi + b\psi).\end{aligned} \tag{2.4}$$

Now by inserting (2.2), (2.3) and (2.4) into (2.1) we get that

$$\mathbb{V}\mathrm{ar}\Big[Z(a\varphi + b\psi) - \big(aZ(\varphi) + bZ(\psi)\big)\Big] = 0,$$

which means that $Z(a\varphi + b\psi) - \big(aZ(\varphi) + bZ(\psi)\big)$ is constant almost surely. By examining the mean we get

$$\mathbb{E}\Big[Z(a\varphi + b\psi) - \big(aZ(\varphi) + bZ(\psi)\big)\Big] = m(a\varphi + b\psi) - m(a\varphi) - bm(a\psi) = 0,$$

and thus $Z(a\varphi + b\psi) - \big(aZ(\varphi) + bZ(\psi)\big) = 0$ almost surely, proving (iii).

To show (iv) we note that if we have sequences of test-functions $\{\varphi_{ik}\}_{k\in\mathbb{N}}$, $i = 1, \ldots, n$, such that $\varphi_{ik} \to \varphi_i$ for $i = 1, \ldots, n$, then

$$\big(m(\varphi_{nk}), \ldots, m(\varphi_{nk})\big)^\top \to \big(m(\varphi_1), \ldots, m(\varphi_n)\big)^\top, \text{ for } k \to \infty,$$

and

$$\begin{bmatrix} C(\varphi_{1k}, \varphi_{1k}) & \cdots & C(\varphi_{1k}, \varphi_{nk}) \\ \vdots & \ddots & \vdots \\ C(\varphi_{nk}, \varphi_{1k}) & \cdots & C(\varphi_{nk}, \varphi_{nk}) \end{bmatrix} \to \begin{bmatrix} C(\varphi_1, \varphi_1) & \cdots & C(\varphi_1, \varphi_n) \\ \vdots & \ddots & \vdots \\ C(\varphi_n, \varphi_1) & \cdots & C(\varphi_n, \varphi_n) \end{bmatrix}, \text{ for } k \to \infty,$$

due to the continuity of m and C. This means that

$$\big(Z(\varphi_{ik}), \ldots, Z(\varphi_{nk})\big)^\top \xrightarrow{d} \big(Z(\varphi_1), \ldots, Z(\varphi_n)\big)^\top, \text{ for } k \to \infty,$$

and thus by Portmanteau's Lemma [Rongfeng, Theorem 1.3] it follows that

$$\mathbb{E}\left[g(Z(\varphi_{ik}), \ldots, Z(\varphi_{nk}))\right] \to \mathbb{E}\left[g(Z(\varphi_1), \ldots, Z(\varphi_n))\right], \text{ for } k \to \infty,$$

for any continuous bounded function $\mathbb{R}^n \to \mathbb{R}$. This shows that Z is continuous by Definition 2.3, and thus it is a generalised stochastic process.

□

We can give meaning to the notion of a derivative of a test function indexed generalised stochastic function, in the following way.

Definition 2.6 (Derivative of a generalised stochastic process). *Let $X(\varphi)$ with $\varphi \in \mathcal{T}_d$ be a generalised stochastic process on a probability space $(\Omega, \mathcal{F}, P)$. Then the generalised stochastic process $-X(\varphi')$ with $\varphi \in \mathcal{T}_d$, is called the derivative of $X(\varphi)$. The derivative is often denoted $X'(\varphi)$.*

Consider now a generalised stochastic process $X(\varphi)$ with $\varphi \in \mathcal{T}_d$ and $\mathbb{E}[X(\varphi)] < \infty$ for all $\varphi \in \mathcal{T}_d$, and recall that the mean value, $m(\cdot)$, is a functional from $\mathcal{T}_d$ to $\mathbb{R}$. Then,

$$\mathbb{E}[X'(\varphi)] = \mathbb{E}[-X(\varphi')] = -m(\varphi') = m'(\varphi), \tag{2.5}$$

where the last equality is a result of (1.4). A similar result is true for the covariance functional, C' of X', as for $\varphi, \psi \in \mathcal{T}_d$, we have

$$\begin{aligned} C'(\varphi, \psi) &= \mathbb{E}[(X'(\varphi) - m'(\varphi))(X'(\psi) - m'(\psi))] \\ &= \mathbb{E}[(X(\varphi') - m(\varphi'))(X(\psi') - m(\psi'))] \\ &= C(\varphi', \psi'). \end{aligned} \tag{2.6}$$

We now turn our attention to an important property of generalised stochastic processes, namely second-order stationarity.

Definition 2.7 (Second-order stationarity). *A generalised stochastic process, Z, is called a second-order stationary stochastic process if*

$$m(\varphi) = m,$$

for all $\varphi \in \mathcal{T}_d$, and there exists a function $\tilde{C}$ such that

$$C(\varphi, \psi) = \tilde{C}(\tau) \quad \text{for all} \quad \varphi, \psi \in \mathcal{T}_d,$$

where $\tau = \varphi - \psi$.

An example of a non-second-order stationary generalised stochastic process is a *Brownian Motion*, which we first define as a stochastic process in the ordinary sense.

Definition 2.8 (Brownian Motion). *Let B_t with $t \in [0, \infty)$ be a stochastic process with*

$$\begin{aligned} \mathbb{E}[B_t] &= 0, \ \text{for all } t \in [0, \infty), \\ \mathbb{C}\text{ov}(B_t, B_s) &= \min(t, s), \ \text{for all } t, s \in [0, \infty). \end{aligned}$$

Then B_t is called a Brownian Motion.

We now represent a Brownian Motion as a generalised stochastic process, defined by

$$B(\varphi, \omega) = \int_0^\infty B_t(\omega)\varphi(t)\mathrm{d}t, \tag{2.7}$$

where B_t is a Brownian motion as defined in Definition 2.8, and $\varphi \in \mathcal{T}_1$. Note that an analogous definition of $B(\varphi)$ is possible for $\varphi \in \mathcal{T}_d$, but for simplicity we focus on the case where $d = 1$. Consider now the mean of $B(\varphi)$, we get

$$\mathbb{E}[B(\varphi)] = \mathbb{E}\left[\int_0^\infty B_t(\omega)\varphi(t)\mathrm{d}t\right] = \int\int_0^\infty B_t(\omega)\varphi(t)\mathrm{d}t\mathrm{d}F_t(x),$$

where $F_t(x)$ is finite dimensional distribution function of B_t. By Tonelli's theorem (see eg [Berg and Madsen]), we may switch the order of integration, yielding

$$\int \int_0^\infty B_t(\omega)\varphi(t)\mathrm{d}t\mathrm{d}F_t(x) = \int_0^\infty \int B_t(\omega)\mathrm{d}F_t(x)\varphi(t)\mathrm{d}t = \int_0^\infty \mathbb{E}[B_t(\omega)]\varphi(t)\mathrm{d}t = 0,$$

for all $\varphi \in \mathcal{T}_1$. The covariance functional is given by

$$C(\varphi,\psi) = \int_0^\infty \int_0^\infty \min(t,s)\varphi(t)\psi(s)\mathrm{d}t\mathrm{d}s.$$

We can now introduce a theorem regarding the derivative of a Brownian Motion.

Theorem 2.9 (Derivative of a Brownian Motion). *Let $B(\varphi)$ with $\varphi \in \mathcal{T}_1$ be a Gaussian generalised stochastic process on a probability space $(\Omega, \mathcal{F}, P)$ with mean value functional*

$$m(\varphi) = 0, \text{ for all } \varphi \in \mathcal{T}_1,$$

and covariance functional

$$C(\varphi,\psi) = \int_0^\infty \int_0^\infty \min(t,s)\varphi(t)\psi(s)\mathrm{d}t\mathrm{d}s, \text{ for all } \varphi,\psi \in \mathcal{T}_1.$$

Then the derivative, $B'(\varphi)$, of $B(\varphi)$ is a Gaussian generalised stochastic process with mean value functional

$$m'(\varphi) = 0, \text{ for all } \varphi \in \mathcal{T}_1,$$

and covariance functional

$$C'(\varphi,\psi) = \int_0^\infty \varphi(t)\psi(s)\mathrm{d}t\mathrm{d}s,$$

where $\varphi,\psi \in \mathcal{T}_1$.

Proof. The following proof is a more detailed and adapted version of the proof in [Schäffler, 2018, Thm 2.4]. Recall that $B'(\varphi) = -B(\varphi')$ by Definition 2.6. Thus, since $B(\varphi)$ is Gaussian, $B'(\varphi)$ is also Gaussian. We now consider the mean and covariance of $B'(\varphi)$. The mean is,

$$m'(\varphi) = -m(\varphi') = 0,$$

by (2.5).
To determine the covariance function of $B'(\varphi)$ we start by considering the covariance function of $B(\varphi)$ which can be separated into two terms as

$$\begin{aligned} C(\varphi,\psi) &= \int_0^\infty \int_0^s t\varphi(t)\psi(s)\mathrm{d}t\mathrm{d}s + \int_0^\infty \int_0^t s\varphi(t)\psi(s)\mathrm{d}s\mathrm{d}t \\ &= \int_0^\infty \psi(s)\int_0^s t\varphi(t)\mathrm{d}t\mathrm{d}s + \int_0^\infty \varphi(t)\int_0^t s\psi(s)\mathrm{d}s\mathrm{d}t. \end{aligned} \tag{2.8}$$

To complete the proof we use integration by parts multiple times, therefore recall that

$$\int_a^b u(t)v'(t)\mathrm{d}t = [v(t)u(t)]_a^b - \int_a^b u'(t)v(t)\mathrm{d}t.$$

Consider now the second term of (2.8). Using integration by parts, and setting $v'(t) = \varphi(t)$ and $u(t) = \int_0^t s\psi(s)\mathrm{d}s$, we have that

$$\int_0^\infty \varphi(t) \int_0^t s\psi(s)\mathrm{d}s\mathrm{d}t = \int_0^\infty u(t)v'(t)\mathrm{d}t = [v(t)u(t)]_0^\infty - \int_0^\infty u'(t)v(t)\mathrm{d}t$$
$$= \int_0^\infty \varphi(s)\mathrm{d}s \int_0^\infty s\psi(s)\mathrm{d}s - \int_0^\infty t\psi(t) \int_0^t \varphi(s)\mathrm{d}s\mathrm{d}t.$$

Defining the function $\Phi(t) = \int_0^t \varphi(s)\mathrm{d}s$, we get

$$\int_0^\infty \varphi(s)\mathrm{d}s \int_0^\infty s\psi(s)\mathrm{d}s - \int_0^\infty t\psi(t) \int_0^t \varphi(s)\mathrm{d}s\mathrm{d}t = \int_0^\infty (\Phi(\infty) - \Phi(t))\psi(t)t\mathrm{d}t.$$

Performing integration by parts analogously on the first term of (2.8) we have

$$C(\varphi, \psi) = \int_0^\infty (\Phi(\infty) - \Phi(t))\psi(t)t + (\Psi(\infty) - \Psi(t))\varphi(t)t\mathrm{d}t,$$

where $\Psi(t) = \int_0^t \psi(s)\mathrm{d}s$. We now use integration by parts once more, with $u(t) = t$ and $v'(t) = (\Phi(\infty)-\Phi(t))\psi(t)+(\Psi(\infty)-\Psi(t))\varphi(t)$. Noting that $v(t) = -(\Phi(\infty)-\Phi(t))(\Psi(\infty)-\Psi(t))$ we have

$$C(\varphi, \psi) = [u(t)v(t)]_0^\infty + \int_0^\infty v(t)u'(t)\mathrm{d}t$$
$$= \lim_{t\to\infty}(u(t)v(t)) + \int_0^\infty v(t)u'(t)\mathrm{d}t$$
$$= \int_0^\infty (\Phi(\infty) + \Phi(t))(\Psi(\infty) - \Psi(t))\mathrm{d}t,$$

where the third equality is a result of $v'(t) \to 0$ as $t \to \infty$.

Recall that $\Phi(t)$ and $\Psi(t)$ are the antiderivative of $\varphi(t)$ and $\psi(t)$, respectively. The covariance functional of $B'(\varphi)$ is then

$$C'(\varphi, \psi) = C(\varphi', \psi') = \int_0^\infty \left(\int_0^\infty \varphi'(s)\mathrm{d}s - \int_0^t \varphi'(s)\mathrm{d}s\right)\left(\int_0^\infty \psi'(s)\mathrm{d}s - \int_0^t \psi'(s)\mathrm{d}s\right)$$
$$= \int_0^\infty (\varphi(\infty) - \varphi(t))\,(\psi(\infty) - \psi(t)) = \int_0^\infty \varphi(t)\psi(t)\mathrm{d}t,$$

where the first equality is due to (2.6), and the last equality is a result of φ and ψ having compact support. This proves the result. □

The derivative of a Brownian Motion constitutes another interesting process, which now formally define for $d = 1$.

Definition 2.10 (Gaussian Discrete White Noise Process). *Let X be a generalised stochastic process with index-set $\mathcal{T}_1$. If $X(\varphi)$ has,*

$$\mathbb{E}[X(\varphi)] = 0, \text{ for all } \varphi \in \mathcal{T}_1,$$

and

$$C(\varphi, \psi) = \int_{-\infty}^\infty \varphi(t)\psi(t)\mathrm{d}t, \text{ for all } \varphi, \psi \in \mathcal{T}_1,$$

then X is called a Gaussian white noise process on $\mathbb{R}$.

A Gaussian white noise process is a generalised stochastic process with independent values. In other words, for two test functions φ and ψ with $\text{supp}(\varphi) \cap \text{supp}(\psi) = \emptyset$,

$$C(\varphi, \psi) = \int_{-\infty}^{\infty} \varphi(t)\psi(t)\mathrm{d}t = 0.$$

A Gaussian white noise process will be used later.

2.2 Generalised Random Fields

This section is based on Mondal [2015].

In this section we introduce the notion of *generalised random fields*. These objects are used to model more complex structures than traditional random fields. When modelling a data-set with random fields, it is assumed that the data is sampled at infinitesimally sized points. Of course this is an abstraction, and in reality the area sampled has a non-zero area. In order to facilitate a model that takes this into account, we introduce set-indexed random fields.

Definition 2.11 (Set-indexed Random Fields). *Let $(X, \mathcal{G}, \mu)$ be measure space and denote $\mathcal{G}_\mu = \{A \in \mathcal{G} : \mu(A) < \infty\}$. Then a stochastic process, Z, with index-set $\mathcal{G}_\mu$, is called a set-indexed random field over $\mathcal{G}$ and μ.*

In order to obtain the situation described above the definition, suppose that Y is an integrable random field on $\mathbb{R}^d$. Then we can define a set-indicated random field over the Borel σ-algebra and the Lebesgue measure as follows. For any bounded set $A \subset \mathbb{R}^d$ we define

$$Z(\omega, A) = \int_A Y(\omega, x)\mathrm{d}x. \tag{2.9}$$

Of course Definition 2.11 also allows other constructions. Henceforth, unless otherwise specified, we only consider set-indicated random fields over the Borel sigma-algebra and the Lebesgue measure. In this project we seek to describe the so-called *De Wijs process*. This can only be defined on *contrasts* of set-indicated random fields. In order to handle this theoretically we need a different class of stochastic processes indexed on measures. First denote $\mathbb{M}(X, \mathcal{G})$ as the set of all signed measures on the measurable space $(X, \mathcal{G})$. Then denote $\mathcal{M} = \{\mu \in \mathbb{M}(\mathbb{R}^d, \mathbb{B}_d) : \mu(\mathbb{R}^d) = 0\}$, where $\mathbb{B}_d$ is the d-dimensional Borel sigma-algebra. It is easy to check that $\mathcal{M}$ is a vector space, which gives rise to the following definition.

Definition 2.12 (Generalised Random Fields). *Let Z be a stochastic process with index-set $\mathcal{S}$. Z is called a generalised random field if $\mathcal{S}$ is a subspace of $\mathcal{M}$, and if it satisfies that*

$$Z(a\mu + b\nu) = aZ(\mu) + bZ(\nu),$$

for all $\mu, \nu \in \mathcal{S}$ and $a, b \in \mathbb{R}$.

A generalised random field can be used to model contrasts over set indexed random fields as follows. We can define a generalised random field as

$$W(\omega, \mu) = \int Y(\omega, x)\mathrm{d}\mu(x).$$

When a generalised random field has this form, we call Y the *underlying random field*. Now let Z be defined as in (2.9) and let $A, B \in \mathbb{R}^d$ with $|A| = |B|$, where $|\cdot|$ is the Lebesgue measure. Suppose μ is a signed measure on $(\mathbb{R}^d, \mathbb{B}_d)$ with density $1_A - 1_B$ with respect to the Lebesgue measure. Then we have

$$\mu(\mathbb{R}^d) = \int 1_A(x)\mathrm{d}x - \int 1_B(x)\mathrm{d}x = |A| - |B| = 0,$$

and thus we get

$$\begin{aligned} W(\omega, \mu) &= \int Y(\omega, x)\mathrm{d}\mu(x) = \int Y(\omega, x)(1_A(x) - 1_B(x))\mathrm{d}x \\ &= \int_A Y(\omega, x)\mathrm{d}x - \int_B Y(\omega, x)\mathrm{d}x = Z(\omega, A) - Z(\omega, B). \end{aligned}$$

If we were to restrict W to the set of measures $\mu \in \mathcal{M}$ with a test-function as density with respect to the Lebesgue measure, we would have a generalised stochastic process where $Z(\mu) = Z(\varphi)$ with an abuse of notation, when φ is the density for μ. That is

$$W(\omega, \varphi) = \int Y(\omega, x)\varphi(x)\mathrm{d}x, \tag{2.10}$$

for all $\varphi \in \mathcal{T}_d$. It is easily seen that the space of measures in $\mathcal{M}$ with a test-function as density with respect to the Lebesgue measure is vector space. Thus it is possible to construct a process that is both a generalised stochastic process and a generalised random field.
The usual covariance function may not be sufficient to model the covariance-structure of generalised random fields. Instead a different but related object is used.

Definition 2.13 (Generalised Covariance Function). *Let Z be a generalised random field, where $\mathbb{E}\left[Z(\mu)\right] = 0$ for all $\mu \in \mathcal{M}$. Then a symmetric function $K(\cdot, \cdot)$ which satisfies*

$$\mathbb{C}\mathrm{ov}\left[Z(\mu), Z(\nu)\right] = \int\int K(x, y)\mathrm{d}\mu(x)\nu(y),$$

is called a generalised covariance function (GCF) for Z.

If a generalised random field is second-order stationary, we can write $K(y - x) := K(x, y)$ with abuse of notation. If furthermore $K(\|x - y\|) = K(x, y)$, then the generalised random field is said to be isotropic.

It turns out that the notion of GCFs for intrinsic random fields of order k, which are examined in Jensen and Fitzhugh [2018], is a special case of Definition 2.13. We can define Λ_k as the space of allowable discrete measures of order k, i.e signed measures on the form

$$\lambda(A) = \sum_{i=1}^{m} \lambda_i \delta_{x_i}(A),$$

where $x_1, \ldots, x_m \in \mathbb{R}^d$ are distinct points, δ_{x_i} is the Dirac-measure centered at x_i and $\lambda_1, \ldots, \lambda_m \in \mathbb{R}$ are chosen such that

$$\sum_{i=1}^{m} \lambda_i P(x_i) = 0,$$

for any polynomial P with $\deg(P) \leq k$. We now define a stochastic process with index-set Λ_k as

$$Z(\lambda) = \sum_{i=1}^{m} \lambda_i \tilde{Z}(x_i),$$

where $\tilde{Z}$ is a random field over $\mathbb{R}^d$. Since Λ_k is a subspace of $\mathcal{M}$, Z is a generalised random field. It is worth noting that the notion of intrinsic random fields of order k defined using random fields over $\mathbb{R}^d$ (see Jensen and Fitzhugh [2018, Definition 1.12]), is equivalent to requiring that Z is stationary. In other words, for $\lambda, \mu \in \Lambda_k$ with $\lambda(A) = \sum_{i=1}^m \lambda_i \delta_{x_i}(A)$ and $\mu(A) = \sum_{i=1}^m \mu_i \delta_{y_i}(A)$ we get that

$$\mathbb{C}\text{ov}\left[Z(\lambda), Z(\mu)\right] = \int\int K(y-x)\mathrm{d}\lambda(y)\mu(x) = \sum_{i=1}^m \sum_{j=1}^m \lambda_i \mu_j K(y_j - x_i).$$

This is equivalent to Jensen and Fitzhugh [2018, Definition 1.16].
In this project we do not prove that the GCF of a generalised random field exists, but for a proof of the special case of a generalised random field with index-set Λ_k and a $k+1$ times mean square differentiable underlying random field over $\mathbb{R}$, see Jensen and Fitzhugh [2018, Theorem 1.18].

Consider the Gaussian Discrete white noise process, as defined in Definition 2.10. We can interpret such a process as a generalised random field, by arguments in the next section, indexed over measures μ with test-functions as density wrt the Lebesgue measure. We can see that such a process has the Dirac delta function as its generalised covariance function, as for a measure μ with density $\varphi(x)$ and another measure ν with density $\psi(y)$ we have

$$\begin{aligned}\mathbb{C}\text{ov}(Z(\mu), Z(\nu)) = \int\int \delta_{x-y}\mathrm{d}\mu(x)\nu(y) &= \int\int \delta_{x-y}\varphi(x)\psi(y)\mathrm{d}x\mathrm{d}y \\ &= \int \varphi(x)\psi(x)\mathrm{d}x\end{aligned}$$

is exactly the covariance function in Definition 2.10. In the above, the second equality is a result of the properties of the dirac delta function, where

$$\int \delta_{x-y}\psi(y)\mathrm{d}y = \psi(x).$$

Relation between index-sets

As we have now seen, we are considering stochastic processes with multiple different types of index-sets. Specifically, we have measures, test-functions and sets. We will now show the relation between these types of index-sets. Before we do this, we introduce the general class of stochastic processes. Note that this class is mostly uninteresting in the context of this project.

Definition 2.14 (Measure indexed stochastic process). *Let $(X, \mathcal{G}, \mu)$ be a measure space. A stochastic process, Z, with index-set of all signed measures on the measurable space $(X, \mathcal{G})$, i.e $\mathbb{M}(X, \mathcal{G})$, is called a Measure indexed stochastic process.*

We can now show that all previously introduced stochastic processes can be presented as measure indexed stochastic processes.

A generalised stochastic process, as defined in Definition 2.4, can be written as a measure indexed stochastic process, as we may consider measures, μ, with test-functions, φ, as densities wrt. the Lebesgue measure. We can then define $Z(\omega, \varphi) = Z(\omega, \mu)$.

For a random field defined on sets $A \subset \mathbb{R}^d$, we may consider this a measure-indexed stochastic process with measures that have the indicator function, $1_A(x)$, as density. In this way, we can define $Z(\omega, A) = Z(\omega, \mu)$.

It is easily seen that generalised random fields are measure indexed stochastic processes, with an additional restriction that the measures, μ, satisfy that $\mu(\mathbb{R}^d) = 0$.

We can now show additional overlaps in the definitions of generalised random fields, generalised stochastic processes, and set-indexed random fields.

Generalised random fields and generalised stochastic process overlap in the following way. Recall that generalised random fields are defined on measures, μ, where $\mu(\mathbb{R}^d) = 0$. A generalised stochastic process can, therefore, be viewed as a generalised random field by restricting the index-set to test-functions which integrates to zero on $\mathbb{R}^d$. Likewise, a generalised random field can be viewed as a generalised stochastic process, by restricting the index-set to measures which have test-functions as density wrt the Lebesgue measure.

Set-indexed random fields and generalised stochastic processes overlap in the following way. We can express a generalised stochastic process as a set-indexed random field by defining (with an abuse of notation) $Z(\omega, \varphi) = Z(\omega, \text{Supp}(\varphi))$, and vice versa.

Lastly, it is possible to express a set-indexed random field as a generalised random field, however, this would be a fairly trivial object, since for a measure with density 1_A with respect to the Lebesgue measure it holds that

$$\mu(\mathbb{R}^d) = \int \mathrm{d}\mu = \int 1_A(x)\mathrm{d}x = |A|.$$

This means that to express a set-indexed random field as a generalised random field, it would require restricting the index-set to only Borel sets with measure zero, that is countable unions of points in $\mathbb{R}^d$.
The relations between the different objects discussed here are summarised in Figure 2.1.

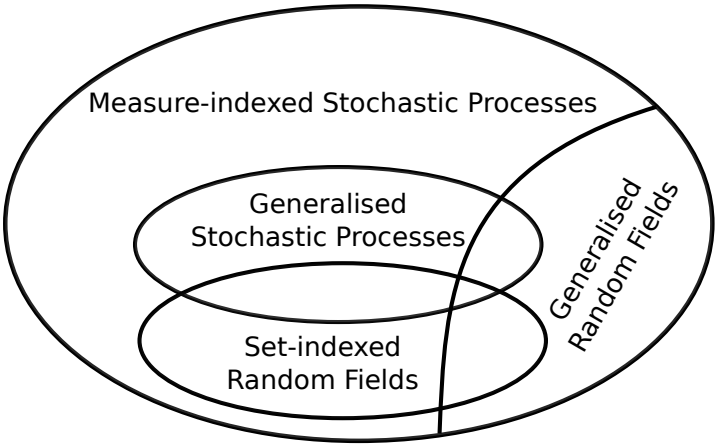


Figure 2.1: Figure showing the relation between measure-indexed stochastic processes, generalised random fields, generalised stochastic processes and set-indexed random fields.

2.3 De Wijs Process

In this section we introduce the process which is of central importance in the thesis. This process is a generalised random field with a particular GCF.

Definition 2.15 (De Wijs Process). *Let Z be a Gaussian generalised random field with index-set $\mathcal{W}$, which is the set of all measures $\mu \in \mathcal{M}$ that satisfy*

$$\int\int |\log\|x-y\|\,|\mathrm{d}\mu^+(x)\mathrm{d}\mu^+(y) < \infty,$$

where $\mu^+(A) = |\mu(A)|$ for all $A \in \mathbb{B}_d$. Z is called the De Wijs process if it is stationary, isotropic, self-similar and has GCF
$K(x,y) = -\log\|x-y\|$, that is

$$\mathbb{C}\mathrm{ov}\left[Z(\mu), Z(\nu)\right] = -\int\int \log\|x-y\|\,\mathrm{d}\mu(x)\mathrm{d}\nu(y),$$

for all $\mu, \nu \in \mathcal{W}$.

Note that the property of self-similarity in this context means that the covariance is preserved under scaling. In some literature, self-similar has a different meaning. We further note that by Definition 2.12 the index set of a generalised random field must be a subspace of $\mathcal{M}$. This means that in order for the De Wijs process to be well defined, we must show that $\mathcal{W}$ is a subspace of $\mathcal{M}$. Obviously $\mathcal{W} \subset \mathcal{M}$, so we need to show that $\mathcal{W}$ is a vector-space. For this purpose we state Mattner [1997, Corollary 2.5], here formulated as a lemma.

Lemma 2.16. *Let f be an infinitely differentiable function such that $(-1)^n f^{(n)}$ is non-constant and non-negative everywhere for $n \geq k$ for some non-negative integer k. Then the set*

$$M^f(\mathbb{R}^d) = \left\{\mu \in M(\mathbb{R}^d) : \int\int |f(\|x-y\|)|\mathrm{d}\mu^+(x)\mathrm{d}\mu^+(y) < \infty, \text{ and } \int x^\alpha \mathrm{d}\mu(x) = 0, \text{ for } |\alpha| \leq k-1\right\}$$

where $M(\mathbb{R}^d)$ is the set of all signed Radon measures on $\mathbb{R}^d$, is a vector space. Furthermore we have that

$$\langle \mu, \nu \rangle = \int\int f(\|y-x\|)\mathrm{d}\mu(x)\mathrm{d}\nu(y)$$

is positive definite on $M^f(\mathbb{R}^d)$, that is $\langle \mu, \mu \rangle > 0$ for all μ where $\mu(A) \neq 0$ for any $A \in \mathbb{B}_d$.

We do not prove this result in this project. We now define the notion of Radon measures for clarity.

Definition 2.17 (Radon measure). *Let $(X, \mathbb{B}_d)$ be a measurable space, where $X \subset \mathbb{R}^d$ and $\mathbb{B}_d$ is the d-dimensional Borel $\sigma-$algebra. A Borel measure $\mu : \mathbb{B}_d \to \mathbb{R}$ is a Radon measure if for all compact subsets $K \subseteq X$, $\mu(K) < \infty$.*

Applying Lemma 2.16 gives the following proposition.

Proposition 2.18. *The De Wijs process is well-defined or, equivalently, $\mathcal{W}$ is a vector space.*

Proof. Let $f(r) = -\log r$. Thus we have that $f(r) < 0$ for $r > 1$, and since

$$f^{(n)} = (-1)^n \frac{(n-1)!}{r^n},$$

when $n \geq 1 = k$, we have that $(-1)^n f^{(n)}(r) > 0$ for all $r \in (0, \infty)$. This means that we can choose $k = 1$ $|\alpha| = 0$, and thus we get $M^f(\mathbb{R}^d) = \mathcal{W}$, which is a vector space by Lemma 2.16. □

By Lemma 2.16 we also get that

$$\langle \mu, \nu \rangle = -\int \int \log\|y - x\| \, \mathrm{d}\mu(x)\mathrm{d}\nu(y)$$

is positive definite. We now state Mattner [1997, Corollary 2.4]

Proposition 2.19. *Let f be an infinitely differentiable function such that $(-1)^n f^{(n)}$ is non-constant and non-negative everywhere for $n \geq k$ for some non-negative integer k, and let μ be a measure in $M(\mathbb{R}^d)$ that is not constantly zero, which satisfies*

$$\int_{\mathbb{R}^d} x^\alpha \mathrm{d}\mu(x) = 0 \tag{2.11}$$

for $|\alpha| \leq k - 1$. Then we get that

$$\int_{\mathbb{R}^d} \int_{\mathbb{R}^d} f(\|x - y\|^2) \mathrm{d}\mu(x)\mathrm{d}\mu(y) > 0. \tag{2.12}$$

We note that the function $-\log(\sqrt{r})$ satisfies the conditions of Proposition 2.19 with $k = 1$. We also note that when $k = 1$ the condition in (2.11) amounts to $\mu(\mathbb{R}^d) = 0$, thus the measures satisfying this condition are all non-zero measures in $\mathcal{M}$. Applying the proposition to $-\log\sqrt{r}$ then implies that $\langle \mu, \mu \rangle = 0$ if and only if $\mu(A) = 0$ for all $A \in \mathbb{B}_d$. This means that $\mathcal{W}$ is a normed inner product space, with norm

$$\|\mu\|_{\mathcal{W}} = \langle \mu, \mu \rangle^{1/2}.$$

It is now straight-forward to show the existence of the de Wijs process. First we note that the covariance-function

$$C(\mu, \nu) = -\int \int \log(\|x - y\|) \mathrm{d}\mu(x)\mathrm{d}\nu(y) \tag{2.13}$$

is positive definite. Thus by using the same arguments as in (i), (ii) and (iii) in the proof of Theorem 2.5 we can show that there exists a generalised random field such that for any $\mu_1, \ldots, \mu_n \in \mathcal{W}$ we have

$$\begin{bmatrix} Z(\mu_1) \\ \vdots \\ Z(\mu_n) \end{bmatrix} \sim N\left(\begin{bmatrix} m(\mu_1) \\ \vdots \\ m(\mu_n) \end{bmatrix}, \begin{bmatrix} C(\mu_1, \mu_1) & \cdots & C(\mu_1, \mu_n) \\ \vdots & \ddots & \vdots \\ C(\mu_n, \mu_1) & \cdots & C(\mu_n, \mu_n) \end{bmatrix} \right),$$

where C is given as in (2.13) and m can be any linear functional which maps form $\mathcal{W}$ to $\mathbb{R}$, and where $Z(a\mu + b\nu) = aZ(\mu) + bZ(\nu)$ almost surely. In order to show that the de Wijs process exists, it suffices to show that C is preserved under translation, rotation and scaling, and that

there exists choices of m that is also preserved under translation, rotation and scaling. Denoting $\mu_h(A) = \mu(A + h)$ for a $h \in \mathbb{R}^d$ we get

$$\begin{aligned} C(\mu_h, \nu_h) &= -\int\int \log\|x - y\| \, \mathrm{d}\mu_h(x)\mathrm{d}\nu_h(y) \\ &= -\int\int \log\|(\tilde{x} - h) - (\tilde{y} - h)\| \, \mathrm{d}\mu_h(\tilde{x} - h)\mathrm{d}\nu_h(\tilde{y} - h) \\ &= -\int\int \log\|\tilde{x} - \tilde{y}\| \, \mathrm{d}\mu(\tilde{x})\mathrm{d}\nu(\tilde{y}) = C(\mu, \nu), \end{aligned}$$

where the second equality comes from performing the change of variable $\tilde{x} = x + h$. Denoting $\mu_R(A) = \mu(RA)$ where R is a rotation matrix, it follows from similar arguments that $C(\mu_R, \nu_R) = C(\mu, \nu)$. Finally we show that the covariance function is invariant under changes of scale. Denoting $\mu_a(A) = \mu(aA)$ for $a \in \mathbb{R}$ we get

$$\begin{aligned} C(\mu_a, \nu_a) &= -\int\int \log\|x - y\| \, \mathrm{d}\mu_a(x)\mathrm{d}\nu_a(y) \\ &= -\int\int \log\|(\tilde{x} - \tilde{y})/a\| \, \mathrm{d}\mu_a(\tilde{x}/a)\mathrm{d}\nu_a(\tilde{y}/a) \\ &= -\int\int \log\|(\tilde{x} - \tilde{y})\| \, \mathrm{d}\mu(\tilde{x})\mathrm{d}\nu(\tilde{y}) + \int\int \log|a|\mathrm{d}\mu(\tilde{x})\mathrm{d}\nu(\tilde{y}) \\ &= C(\mu, \nu), \end{aligned}$$

where the second equality comes from performing the change of variable $\tilde{x} = ax$ and the final equality comes from the fact that

$$\int\int \log|a|\mathrm{d}\mu(\tilde{x})\mathrm{d}\nu(\tilde{y}) = \log|a| \int\int \mathrm{d}\mu(\tilde{x})\mathrm{d}\nu(\tilde{y}) = 0,$$

since

$$\int \mathrm{d}\mu(\tilde{x}) = \mu(\mathbb{R}^d) = 0.$$

We have now shown that there exists a Gaussian generalised random field with the same covariance functional as the de Wijs process, and that this covariance functional is invariant under translation, rotations and scaling. In order for such a generalised random field to be the de Wijs process, it must be stationary, isotropic and self-similar. Since all the finite dimensional distributions are Gaussians it is sufficient to show that there exists a valid mean functional which is invariant under translations, rotations and scaling. This is trivial since we may choose

$$m(\mu) = 0, \text{ for all } \mu \in \mathcal{W}.$$

Lastly we note that we cannot choose the mean functional to be constantly non-zero. To see this we note the requirement of linearity of the mean functional. Now suppose that $m(\mu) = c \neq 0$ for all $\mu \in \mathcal{W}$ and choose $a, b \in \mathbb{R}$ such that $a + b \neq 1$. Then we get

$$c = m(a\mu + b\nu) = am(\mu) + bm(\nu) = (a + b)c,$$

which is a contradiction. Thus a constantly non-zero mean functional is not a valid choice of mean functional.

We can investigate when a symmetric function, K, is a valid GCF. Recall from Definition 2.13 that a symmetric function K is called a GCF for a generalised random field Z, if it satisfies that

$$\mathbb{C}\mathrm{ov}(Z(\mu), Z(\nu)) = \int\int K(x, y)\mathrm{d}\mu\mathrm{d}\nu.$$

Let $Y = \sum_{i=1}^{k} a_i Z(\mu_i)$, where μ_i are measures such that $\mu_i(\mathbb{R}^d) = 0$ and $a_i \in \mathbb{R}$ for all i. Then we have

$$\mathbb{V}\mathrm{ar}(Y) = \mathbb{V}\mathrm{ar}\left(\sum_{i=1}^{k} a_i Z(\mu_i)\right) = \sum_{i=1}^{k}\sum_{j=1}^{k} a_i a_j \mathbb{C}\mathrm{ov}(Z(\mu_i), Z(\mu_j)) \geq 0.$$

Using the definition of the GCF we furthermore get

$$\sum_{i=1}^{k}\sum_{j=1}^{k} a_i a_j \int\int K(x,y)\mathrm{d}\mu_i \mathrm{d}\mu_j = \int\int K(x,y) \sum_{i=1}^{k}\sum_{j=1}^{k} a_i a_j \mathrm{d}\mu_i \mathrm{d}\mu_j.$$

Denoting $\tilde{\mu_1}(x) = \sum_{i=0}^{k} a_i \mu_i(x)$ and $\tilde{\mu_2}(y) = \sum_{j=0}^{k} a_i \mu_i(y)$ we have the requirement that the GCF, K, satisfies

$$\int\int K(x,y)\mathrm{d}\tilde{\mu_1}\mathrm{d}\tilde{\mu_2} \geq 0.$$

It is possible to determine if a given parametrisation of K satisfies this criterion, using Lemma 2.16.

2.4 Restricted Maximum Likelihood Estimation of Covariance Structure

In this section, we describe a method for estimating the covariance structure of a generalised random field. In particular, we wish to derive the likelihood for the so-called De Wijs plus white noise process, which we first formally define.

Definition 2.20 (The De Wijs plus White Noise Process). *Let Z be a generalised random field. Suppose Z has generalised covariance function*

$$K(x-y) = \sigma_0^2 \delta_{x-y} - \sigma_1^2 \log(\|x-y\|),$$

where $\sigma_0, \sigma_1 > 0$, and δ is Diracs delta function. Then Z is called the De Wijs plus white noise process.

A model of the GCF, as in Definition 2.20, which is a combination of two GCF models (white noise and De Wijs) is typically refered to as a *conformal* model in the literature(see e.g. [Clifford and McCullagh, 2006]).

Suppose now that, we have observed n data points indexed by sets, $A_i \subset \mathbb{R}^d$ for $i = 1, \dots, n$. We assume that such data is an average of a random field in a specific area, i.e. we assume that we observe

$$Y(A_i) = \int_{A_i} Y(x) \mathrm{d}x.$$

To model such data using the De Wijs plus white noise process, we must find a way to interpret the data as a realisation of a generalised random field. Recall that such random fields are indexed using measures, μ, where $\mu(\mathbb{R}^d) = 0$. We make this interpretation using contrasts of the observed data, such that we interpret

$$Z(\mu_i) = Y(A_i) - Y(A_1) = \int Y(x)[1_{A_{i+1}} - 1_{A_1}](x)\mathrm{d}x = \int Y(x)\mathrm{d}\mu_i(x),$$

where $i = 2, \dots, n$ as $i = 1$ is uninteresting. Note that in this section, we work with contrasts of the type above, but more general contrasts are also an option. Note that it is necessary to assume that $Y(x)$ has a constant mean, as Z must have zero mean by the arguments of the last section. We furthermore assume that Y is Gaussian.
We now have a generalised random field Z indexed over measures μ_i for $i = 2, \dots, n$, where μ_i has the density $1_{A_i}(x) - 1_{A_1}(x)$ wrt the Lebesgue measure. Notice that working with contrasts, in the way above, where the mean structure is eliminated, is reminiscent of *Restricted Maximum Likelihood* (REML) estimation, which we now briefly present in a general setting.

Restricted Maximum Likelihood Estimation in General

The main of idea of REML estimation is to apply a linear transformation to data, in order to eliminate the mean value structure. In other words, let Y be an $n \times 1$ stochastic vector, with $\mathbb{E}[Y] = \mu \in \mathrm{span}(X)$. Now let A be an $(n-m) \times n$ matrix with columns that span the orthogonal

complement of span(X). Considering the matrix multiplication $Z = AY$, we have that

$$\mathbb{E}[Z] = A\mathbb{E}[Y] = 0$$
$$\mathbb{C}\text{ov}(Z) = A\mathbb{C}\text{ov}(Y)A^\top.$$

After the linear transformation is applied, we perform ordinary maximum likelihood estimation of the parameters. Typically, one would write up the profile log-likelihood instead of the ordinary likelihood, eliminating parameters, not of interest. A strength of REML estimation is that it produces unbiased estimates.

2.4.1 REML for the De Wijs plus White Noise process

In this subsection, we derive the log-likelihood of the De Wijs plus white noise process as defined in Definition 2.20.
Recall that we interpret contrasts of data, as realisations of a zero-mean Gaussian generalised random field indexed using measures μ_i which have density $1_{A_i}(x) - 1_{A_1}(x)$ wrt. the Lebesgue measure for $i = 2, \ldots, n$. We now show that this setup results in a positive definite covariance matrix. Note that other contrasts are possible and that this will be addressed later.

Positive Definiteness of the covariance matrix

Since Z is Gaussian, we can write up its likelihood, by computing the covariance matrix. However, for the likelihood to be defined, the covariance matrix must be positive definite, and thus invertible. To show that this is the case for the De Wijs plus white noise process, we must show that

$$a^\top \mathbb{C}\text{ov}(Z)a = \sum_{i=2}^{n}\sum_{j=2}^{n} a_i a_j \mathbb{C}\text{ov}(Z(\mu_i), Z(\mu_i)) > 0$$

Specifically for the De Wijs plus white noise process we have by Definition 2.20 that

$$\mathbb{C}\text{ov}(Z(\mu_i), Z(\mu_j)) = \sigma_0^2 \int\int \delta_{x-y}\mathrm{d}\mu_i\mathrm{d}\mu_j - \sigma_1^2 \int\int \log(\|x-y\|)\mathrm{d}\mu_i\mathrm{d}\mu_j. \tag{2.14}$$

Therefore we must show that,

$$\sum_{i=2}^{n}\sum_{j=2}^{n} a_i a_j \sigma_0^2 \int\int \delta_{x-y}\mathrm{d}\mu_i\mathrm{d}\mu_j - \sum_{i=2}^{n}\sum_{j=2}^{n} a_i a_j \sigma_1^2 \int\int \log(\|x-y\|)\mathrm{d}\mu_i\mathrm{d}\mu_j > 0$$

We can explicitly compute the first term of (2.14), using the densities of the measure. First note that the Dirac delta function has the useful property that,

$$\int_{A_j} \delta_{x-y}\mathrm{d}x = 1_{A_j}(y).$$

We therefore have that

$$\begin{aligned}\sigma_0^2 \int\int \delta_{x-y}\mathrm{d}\mu_i\mathrm{d}\mu_j = \sigma_0^2\Big(&\int_{A_i}\int_{A_j} \delta_{x-y}\mathrm{d}x\mathrm{d}y - \int_{A_1}\int_{A_j} \delta_{x-y}\mathrm{d}x\mathrm{d}y \\ &- \int_{A_i}\int_{A_1} \delta_{x-y}\mathrm{d}x\mathrm{d}y + \int_{A_1}\int_{A_1} \delta_{x-y}\mathrm{d}x\mathrm{d}y\Big) \\ &= \sigma_0^2\left(|A_i \cap A_j| - |A_1 \cap A_j| - |A_i \cap A_1| + |A_i|\right),\end{aligned}$$

where the second and third terms are zero, since $i, j \neq 1$. Writing the above on matrix form, and recalling that $i, j = 2, \ldots, n$, we obtain the matrix

$$\sigma_0^2 |A_i| \left(I_{(n-1)} + 1_{(n-1)} 1_{(n-1)}^\top \right),$$

which is a positive definite matrix, because the addition in the parentheses results in a positive definite matrix.

It remains to be shown that

$$-\sum_{i=2}^{n}\sum_{j=2}^{n} a_i a_j \sigma_1^2 \int\int \log(\|x-y\|) \mathrm{d}\mu_i \mathrm{d}\mu_j > 0.$$

This can be shown by the arguments from Section 2.3, which we outline here again. We may switch the summation and integration such that

$$-\sum_{i=2}^{n}\sum_{j=2}^{n} a_i a_j \sigma_1^2 \int\int \log(\|x-y\|) \mathrm{d}\mu_i \mathrm{d}\mu_j = \int\int -\log(\|x-y\|) \sum_{i=2}^{n}\sum_{j=2}^{n} a_i a_j \mathrm{d}\mu_i(x) \mathrm{d}\mu_j(y).$$

Denoting $\tilde{\mu}(x) = \sum_{i=2}^{n} a_i \mu_i(x)$ and $\tilde{\mu}(y) = \sum_{i=2}^{n} a_j \mu_j(y)$ we have

$$\int\int -\log(\|x-y\|) \sum_{i=2}^{n}\sum_{j=2}^{n} a_i a_j \mathrm{d}\mu_i(x) \mathrm{d}\mu_j(y) = \int\int -\log(\|x-y\|) \tilde{\mu}(x) \tilde{\mu}(y) > 0,$$

by Lemma 2.16, since $\tilde{\mu}(x)$ and $\tilde{\mu}(y)$ are Radon measures (see Definition 2.17). Thus the matrix $\mathbb{C}\mathrm{ov}(Z)$ is positive definite.

Computing the covariance matrix

Having shown the positive definiteness of $\mathbb{C}\mathrm{ov}(Z)$ we now compute it explicitly. Recall from earlier that

$$\mathbb{C}\mathrm{ov}(Z(\mu_i), Z(\mu_j)) = \sigma_0^2 \left(|A_i \cap A_j| + |A_i| \right) - \sigma_1^2 \int\int \log(\|x-y\|) \mathrm{d}\mu_i \mathrm{d}\mu_j, \tag{2.15}$$

for $i, j = 2, \ldots, n$. We now consider the second term. In the following we use the notation

$$I_{ij} = \int_{A_i}\int_{A_j} \log(\|x-y\|) \mathrm{d}x \mathrm{d}y.$$

Using the densities of the index measures, we have

$$-\sigma_1^2 \int\int \log(\|x-y\|) \mathrm{d}\mu_i \mathrm{d}\mu_j = \sigma_1^2 (-I_{ij} + I_{1j} + I_{i1} - I_{11}). \tag{2.16}$$

We can thus write the covariance matrix of Z as

$$\mathbb{C}\mathrm{ov}(Z) = \sigma_0^2 |A_i| \left(I + 11^\top \right) + \sigma_1^2 V, \tag{2.17}$$

where the entrances of V are

$$V_{i,j} = -I_{ij} + I_{1j} + I_{i1} - I_{11}$$

To compute the matrix V, we thus need to compute I_{ij} for $i, j = 1, \ldots, n$. There are a variety of ways to approximate I_{ij}, and we show two different approaches in the next subsection.

Approximating integrals over subsets of $\mathbb{R}^d$

In this subsection we present different approaches in approximating the integral

$$\int_{A_i}\int_{A_j} \log(\|x-y\|)\mathrm{d}x\mathrm{d}y, \tag{2.18}$$

where A_i and A_j for $i,j=1,\ldots,n$, are subsets of $\mathbb{R}^d$.

The first approach is reminiscent of Riemann approximation. We split the computation into two cases when $i=j$ and when $i\neq j$.
For $i\neq j$ we approximate the integral by

$$\int_{A_i}\int_{A_j} \log(\|x-y\|)\mathrm{d}x\mathrm{d}y = |A_i||A_j|\log(\|x_i-y_j\|),$$

where x_i and y_i are representative points in A_i and A_j, respectively. This approach is more troublesome when $i=j$, as the negative logarithm tends to infinity as the distance between representative points is small. Instead we consider the area A_i, and take the representative points x_i and y_i to be points with the furthest possible distance in A_i We denote the logarithm of this distance as k. Note that this quite possibly results in a small approximation of the integral. Using this approximation the entrances of the matrix V become

$$V_{ij} = -|A_i||A_j|\log(\|x_i-y_j\|) + |A_i|^2\left(\log(\|x_1-y_j\|)+\log(\|x_i-y_1\|)\right) - |A_1|^2 k.$$

The diagonal of V is given by

$$V_{ii} = -2|A_i|^2 k + 2|A_1||A_i|\log(\|x_1-y_i\|).$$

The second approach of approximating the integral, is using Monte Carlo methods. Consider the integral in (2.18), and assume that $p(x)$ and $q(y)$ are the densities of uniform distributions in A_i and A_j, respectively. Multiplying and dividing, we have

$$\int_{A_i}\int_{A_j} \log(\|x-y\|)\mathrm{d}x\mathrm{d}y = \int_{A_i}\int_{A_j} \frac{\log(\|x-y\|)}{p(x)q(y)}p(x)q(y)\mathrm{d}x\mathrm{d}y.$$

We can thus perceive the integral as

$$\mathbb{E}\left[\frac{\log(\|x-y\|)}{p(x)q(y)}\right] = |A_i||A_j|\mathbb{E}[\log(\|x-y\|)],$$

since the uniform densities are simply the reciprocal of the volume of the area in question. We can approximate the mean value by simulating n points, x_i, in A_i and, y_i in A_j, uniformly. Afterwards, we approximate the mean value using the empirical mean, i.e we use the approximation

$$\int_{A_i}\int_{A_j} \log(\|x-y\|)\mathrm{d}x\mathrm{d}y \approx |A_i||A_j|\frac{1}{n}\sum_{i=1}^{n}\log(\|x_i-y_i\|). \tag{2.19}$$

Since this is an approximation, it would be useful to asses the accuracy of the approximation. This can be done by computing the empirical standard error of the Monte Carlo estimate. Denoting the estimate in (2.19) as Q, we can compute

$$s^2 = \frac{1}{n-1}\sum_{i=1}^{n}(|A_i||A_j|\log(\|x_i-y_i\|)-Q)^2.$$

Then the variance of the estimate is approximately,

$$\mathbb{V}\text{ar}(Q) \approx \frac{s^2}{n}.$$

We can then compare the size of the estimate with the empirical variance to asses the accuracy of the estimate.

Likelihood of the covariance parameters

Recall that Z is Gaussian, and since the covariance matrix is positive definite, Z has a Gaussian likelihood. Let z be a realisation of Z and let Ψ be the covariance matrix in (2.17), then since $\mathbb{E}[Z] = 0$ the log-likelihood is

$$l(\Psi) = -\frac{1}{2}\log(|\Psi|) - \frac{1}{2}z^\top \Psi^{-1} z \tag{2.20}$$

As the covariance matrix Ψ of Z is a function of σ_0^2 and σ_1^2, we have

$$l(\sigma_0^2, \sigma_1^2) = -\frac{1}{2}\log(|\Psi(\sigma_0^2, \sigma_1^2)|) - \frac{1}{2}z^\top \Psi(\sigma_0^2, \sigma_1^2)^{-1} z. \tag{2.21}$$

It is possible to derive a profile log-likelihood by considering a ratio of the parameters. We can re-parametrise the covariance matrix in (2.17), as

$$\mathbb{C}\text{ov}(Z) = \sigma_0^2 |A_i| \left(I + 11^\top\right) - \sigma_1^2 V = \sigma_0^2 \left(|A_i| \left(I + 11^\top\right) - \frac{\sigma_1^2}{\sigma_0^2} V\right) = \sigma_0^2 \Phi(\tau), \tag{2.22}$$

where $\tau = \frac{\sigma_1^2}{\sigma_0^2}$. Inserting the re-parametrisation in (2.20), we get the log-likelihood

$$l(\sigma_0^2, \tau) = -\frac{n-1}{2}\log(\sigma_0^2) - \frac{1}{2}\log(|\Phi(\tau)|) - \frac{1}{2\sigma_0^2} z^\top \Phi(\tau)^{-1} z. \tag{2.23}$$

We can find the profile log-likelihood of τ by maximising the log-likelihood wrt σ_0^2. The partial derivative wrt σ_0^2 is

$$\frac{\partial}{\partial \sigma_0^2} l(\sigma_0^2, \tau) = -\frac{n-1}{2\sigma_0^2} + \frac{1}{2\sigma_0^4} z^\top \Phi(\tau)^{-1} z. \tag{2.24}$$

Setting (2.24) equal to zero, and isolating σ_0^2 we get the estimator

$$\hat{\sigma}_0^2(\tau) = \frac{1}{n-1} z^\top \Phi(\tau)^{-1} z. \tag{2.25}$$

Plugging the estimator back into (2.23), we get the profile log-likelihood for τ

$$\begin{aligned} l(\hat{\sigma}_0^2(\tau), \tau) &= -\frac{n-1}{2}\log\left(\frac{1}{n-1} z^\top \Phi(\tau)^{-1} z\right) - \frac{1}{2}\log(|\Phi(\tau)|) - \frac{1}{2}\frac{n-1}{1}\frac{1}{z^\top \Phi(\tau)^{-1} z} z^\top \Phi(\tau)^{-1} z \\ &= -\frac{n-1}{2}\log\left(\frac{1}{n-1} z^\top \Phi(\tau)^{-1} z\right) - \frac{1}{2}\log(|\Phi(\tau)|) - \frac{n-1}{2}. \end{aligned}$$

As the last term is constant, we get that the profile log-likelihood of τ is proportional to

$$l(\hat{\sigma}_0^2(\tau), \tau) \propto -\frac{n-1}{2}\log\left(\frac{1}{n-1} z^\top \Phi(\tau)^{-1} z\right) - \frac{1}{2}\log(|\Phi(\tau)|), \tag{2.26}$$

which can be maximized to find the estimate $\hat{\tau}$, and thereafter $\hat{\sigma}_0^2$ by utilising (2.25).

The Gaussian Generalised Covariance Function

While the De Wijs plus white noise process, is of central importance is this thesis, it is of course possible to apply alternative parametrisations of the GCF. The simplest choice of model for the GCF is the so-called Gaussian GCF, where the GCF is modelled by the isotropic function

$$K(\|x-y\|) = e^{-a\|x-y\|}.$$

This model depends only on a single positive parameter $a \in \mathbb{R}$.
To determine if the function satisfies the conditions in Lemma 2.16, we compute the partial derivatives. Let $t = \|x-y\|$, then

$$K'(t) = e^{-at}(-a)$$
$$K^{(2)}(t) = e^{-at}a^2$$

For arbitrary order, n, of differentiation we have

$$\frac{d^n}{dt^n}K(t) = e^{-at}(-1)^n a^n.$$

Therefore, we have that for all positive integers n,

$$(-1)^n \frac{d^n}{dt^n}K(t) = e^{-at}(-1)^{2n}a^n > 0, \text{ for all } t \in \mathbb{R}, a > 0,$$

thus satisfying the conditions Lemma 2.16 and constituting a valid model for the GCF of a generalised random field, Z. Note that the Gaussian GCF is the name of the GCF, and does not imply that the generalised random fields in question needs to be Gaussian.

2.5 Alternative choice of contrasts

In Section 2.4, we introduce REML estimation of a De Wijs plus white noise process. We do this for a specific choice of contrasts of data, namely subtracting a specific value from the remaining data. This type of contrast was merely an arbitrary choice, and other contrasts are possible. In this section, we show that other contrasts result in a positive definite covariance matrix.

Recall from the start of Section 2.4 that we have observed n data-points interpreted to be from the set-indexed random field

$$Y(A_i) = \int_{A_i} Y(x)\mathrm{d}x.$$

To be able to model such data, using generalised random fields, we interpret said generalised random field, as contrasts of the set-indexed random field. Note that the sets all have the same volume. Let $x_1, \ldots, x_M \in \mathbb{R}^n$, such that $1_n^\top x_i = 0$ for $i = 1, \ldots, M$. We can define measures

$$\mu_k(B) = \int_B \sum_{i=1}^n (x_k)_i 1_{A_i}(x)\mathrm{d}x, \text{ where } k = 1, \ldots, M.$$

We see that these measures satisfy

$$\mu_k(\mathbb{R}^d) = \int_{\mathbb{R}^d} \sum_{i=1}^n (x_k)_i 1_{A_i}(x)\mathrm{d}x = \sum_{i=1}^n (x_k)_i |A_i| = |A| \sum_{i=1}^n (x_k) = 0, \text{ for } k = 1, \ldots, M,$$

We note that these are Radon measures. Let Z be generalised random field with GCF, $K(x,y)$, and these measures as indices. Then, by Definition 2.13, the covariance is

$$\mathbb{C}\text{ov}(Z(\mu_i), Z(\mu_j)) = \int\int K(x,y)\mathrm{d}\mu_i\mathrm{d}\mu_j, \tag{2.27}$$

for $i = 1,\ldots,M$ and $j = 1,\ldots,M$. Suppose now that Z is a De Wijs plus white noise process, i.e

$$K(x,y) = \sigma_0^2\delta_{x-y} - \sigma_1^2\log(\|x-y\|).$$

Inserting into (2.27), we have

$$\mathbb{C}\text{ov}(Z(\mu_i), Z(\mu_j)) = \sigma_0^2\int\int \delta_{x-y}\mathrm{d}\mu_i\mathrm{d}\mu_j + \sigma_1^2\int\int -\log(\|x-y\|)\mathrm{d}\mu_i\mathrm{d}\mu_j.$$

Using the densities of the measures, we can rewrite the above covariance as

$$\begin{aligned}\mathbb{C}\text{ov}(Z(\mu_i), Z(\mu_j)) = \sigma_0^2\int\int \delta_{x-y}\left(\sum_{m=1}^n (x_i)_m 1_{A_m}(x)\right)\left(\sum_{k=1}^n (x_j)_k 1_{A_k}(y)\right)\mathrm{d}x\mathrm{d}y \\ +\sigma_1^2\int\int -\log(\|x-y\|)\left(\sum_{m=1}^n (x_i)_m 1_{A_m}(x)\right)\left(\sum_{k=1}^n (x_j)_k 1_{A_k}(y)\right)\mathrm{d}x\mathrm{d}y\end{aligned}$$

Using the notation

$$D_{mk} = \int_{A_m}\int_{A_k}\delta_{x-y}\mathrm{d}x\mathrm{d}y,$$

and

$$L_{mk} = \int_{A_m}\int_{A_k}\log(\|x-y\|)\mathrm{d}x\mathrm{d}y,$$

we can further rewrite the covariance as

$$\mathbb{C}\text{ov}(Z(\mu_i), Z(\mu_j)) = \sigma_0^2\sum_{m=1}^n\sum_{k=1}^n (x_i)_m(x_j)_k D_{mk} - \sigma_1^2\sum_{m=1}^n\sum_{k=1}^n (x_i)_m(x_j)_k L_{mk}. \tag{2.28}$$

This implies that the covariance matrix of Z is

$$\mathbb{C}\text{ov}(Z) = \sigma_0^2 TDT^\top + \sigma_1^2 TLT^\top, \tag{2.29}$$

where the matrix L is matrix containing the values $-L_{mk}$, and T is a $M\times n$ matrix whose rows are the vectors x_i for $i = 1,\ldots,M$. Furthermore, the matrix D contains the values d_{mk}. Note that $D = |A|I$, where $|A|$ is the common area of the sets A_i, since

$$d_{mk} = |A_m \cap A_k| = \begin{cases}|A|, \text{ when } m = k.\\ 0, \text{ when } m \neq k.\end{cases}$$

Consider now the second term of (2.29). For this term to be positive definite, we require that the matrix T has full rank. This implies that M is at most $n-1$, as T must be span the null space of the vector 1_n. Coincidently, this also ensures that the first term is positive definite, as the identity matrix is positive definite. We already know that the matrix $TLT^\top$ is positive definite, which is a result of the arguments in Section 2.4, and of Lemma 2.16.

We have thus shown that other contrasts also result in positive definite covariance-matrices. The contrasts must only be orthogonal on the vector 1_n. Note that the REML estimates are invariant under a different choice of contrasts. The invariance is because if both T and T' both span the orthogonal complement to the vector 1_n, then there exists an invertible matrix P such that $T = T'P$. For more details on the invariance of REML estimates under different contrasts, see Waagepetersen [2019].

2.6 Variogram Estimation of covariance parameters

In this section, we describe an estimation procedure of the covariance structure, using the so-called variogram and semi-variogram.
A variogram is an object, which is typically of central importance in practical analysis of ordinary random fields. We now define the object for generalised random fields.

Definition 2.21 (Variogram of Generalised Random Fields). *Let Z be a generalised random field with index set $A \subseteq \mathcal{M}$. The variogram of Z is*

$$2\gamma(\mu,\nu) = \mathbb{V}\mathrm{ar}\left[Z(\mu) - Z(\nu)\right],$$

for $\mu, \nu \in A$.

The function γ is called the semi-variogram. For a given GCF, $K(x,y)$, of a generalised random field Z, the variogram can be written as

$$2\gamma(\mu,\nu) = \mathbb{V}\mathrm{ar}\left[Z(\mu) - Z(\nu)\right] = \mathbb{C}\mathrm{ov}(Z(\mu),Z(\mu)) + \mathbb{C}\mathrm{ov}(Z(\nu),Z(\nu)) - 2\mathbb{C}\mathrm{ov}(Z(\mu),Z(\nu)).$$

Using Definition 2.13, we have

$$2\gamma(\mu,\nu) = \int\int K(x,y)\mathrm{d}\mu\mathrm{d}\mu + \int\int K(x,y)\mathrm{d}\nu\mathrm{d}\nu - 2\int\int K(x,y)\mathrm{d}\mu\mathrm{d}\nu. \tag{2.30}$$

Now let Z be an generalised random field, with index set as in Section 2.4, and GCF

$$K(x,y) = \sigma_0^2\delta_{x-y} - \sigma_1^2\log(\|x-y\|).$$

We can compute the variogram of Z explicitly using (2.30). Recall from Section 2.4, that the GCF above yields the covariance,

$$\mathbb{C}\mathrm{ov}(Z(\mu_i),Z(\mu_j)) = \sigma_0^2\left(|A_i \cap A_j| + |A_i|\right) - \sigma_1^2\int\int \log(\|x-y\|)\mathrm{d}\mu_i\mathrm{d}\mu_j. \tag{2.31}$$

Recall, earlier we use the notation

$$\int_{A_i}\int_{A_j} \log(\|x-y\|)\mathrm{d}x\mathrm{d}y = I_{ij}.$$

We note that $I_{ii} = I_{jj}$ for $i,j = 1,\ldots,n$, and we denote this value M. By (2.30), and using (2.31), the variogram for Z is

$$\begin{aligned}2\gamma(\mu_i,\mu_j) = 2\sigma_0^2|A_i| - \sigma_1^2\left(M - I_{1i} - I_{i1} + M\right) + 2\sigma_0^2|A_j| - \sigma_1^2\left(M - I_{1j} - I_{j1} + M\right)\\ - 2\left[\sigma_0^2|A_i| - \sigma_1^2\left(I_{ij} - I_{1i} - I_{j1} + M\right)\right]\end{aligned}$$

Expanding the above yields

$$\begin{aligned}2\gamma(\mu_i,\mu_j) &= 2\sigma_0^2|A_i| + 2\sigma_1^2 I_{ij} + 2\sigma_1^2 I_{1j} + 2\sigma_1^2 I_{i1} + 2\sigma_1^2 I_{11}\\ &\quad - \sigma_1^2 I_{ii} - \sigma_1^2 I_{1i} - \sigma_1^2 I_{i1} - \sigma_1^2 I_{11} - \sigma_1^2 I_{jj} - \sigma_1^2 I_{1j} - \sigma_1^2 I_{j1} - \sigma_1^2 I_{11}\\ &= 2\sigma_0^2|A_i| + 2\sigma_1^2(I_{ij} - M).\end{aligned}$$

Since the areas A_i are identical for all i, the integral I_{ij} is only dependent on the difference vector between the sets A_i and A_j. In Section 2.4 we briefly discussed ways of approximating I_{ij}. Suppose we use the first discussed approach. Then we get the variogram

$$2\gamma(\mu_i, \mu_j) = 2\sigma_0^2|A_i| + 2\sigma_1^2(|A_i||A_j| \log(\|x_i - y_j\|) - M),$$

where x_i and y_j are representative points in the sets A_i and A_j, respectively. This can be seen as a function of the distance between x_i and y_i. We therefore have the variogram as the function

$$2\gamma(h) = 2\sigma_0^2|A| + 2\sigma_1^2(|A|^2 \log(h) - M), \tag{2.32}$$

where $|A|$ is the volume of the sets $|A_i|$ for all i. Note that the value of M can be estimated using Monte Carlo methods, as discussed in Section 2.4.

We have now used the relation between the GCF and the variogram to compute what we will call a variogram model, i.e. the function in (2.32). It is then possible to estimate the covariance parameters by fitting the variogram model to the empirical variogram. Suppose that the generalised random field Z is indexed using the measures μ_i as in Section 2.4. Assume furthermore that there is an underlying set indexed random field, Y, and we have n observations, $Y(A_i)$ for $i = 1, \ldots, n$. We then see that the variogram of Z is

$$2\gamma(\mu_i, \mu_j) = \mathbb{V}\text{ar}\left[Z(\mu_i) - Z(\mu_j)\right] = \mathbb{V}\text{ar}\left[Y(A_i) - Y(A_j)\right] = \mathbb{E}[(Y(A_i) - Y(A_j))^2],$$

Since the variogram model in (2.32) is only dependent on the distance between A_i and A_j, the same must be true of the mean square difference of Y. In order to compute the variogram for a given distance h, we introduce the set $S(h)$ as the set of pairs (i, j) where $\|x_i - y_j\| = h$, where x_i and y_j are representative points of the sets A_i and A_j, respectively, and where both $Y(A_i)$ and $Y(A_j)$ have been observed. Using this set we can estimate the empirical variogram as

$$\hat{\gamma}(h) = \frac{1}{|S(h)|} \sum_{(i,j)\in S(h)} (Y(A_i) - Y(A_j))^2.$$

Note that $\hat{\gamma}(0) = 0$. After computing the empirical variogram, it is possible to fit the parameters of the variogram model in (2.32), using a goodness-of-fit criteria as, for example, least squares.

Chapter 3

Implementation

3.1 Data Introduction

In this project, we define the De Wijs plus white noise process and seek to apply this model to a particular dataset. The dataset in question contains measurements of the contents in soil from Barro Colorado Island in Panama. Specifically, the contents are aluminium, boron, calcium, copper, iron, potassium, magnesium, manganese, sodium, phosphorus, sulphur, zinc, ammonium, nitrate, nitrogen, mineralised ammonium, mineralised nitrate, mineralised nitrogen and the pH-value of the soil.

A location for each measurement is known, and we call these centre points. The sample points are mainly located on a regularly spaced grid, with a spacing of 50 meters. However, sample points between the grid points are also used, resulting in 300 observation in the dataset. The locations of the samples can be seen in the plot below.

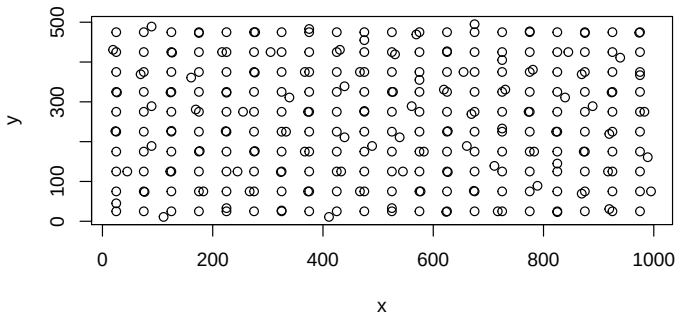


Figure 3.1: Locations of the measurements of soil data.

At each sampling point, three different samples are obtained, all within a radius of 1 meter of each centre point. Note that the exact locations of the three samples are unknown. Each sample is obtained using a so-called LaMotte soil sampling tube [Harms et al., 2004], which is a round tube, extracting a cylinder-shaped sample with a diameter of 2.5 cm and a depth of 30.5 cm. The three samples have a total volume of 0.00044 m^3. Since the exact locations of three samples are unknown, we can approximate the sample area by a single pillar with the same volume. This pillar can have the same height of 30.5 cm, and the side lengths must be approximately

3.8 cm to preserve the volume. These pillars are centred at the centre points. For more detailed information about how the data has been obtained, see Harms et al. [2004].

In the following sections, this dataset is referred to as the `bcicov`-dataset, and the data are used as observations of the underlying random field, Y, where the generalised random field, Z, is interpreted as contrasts hereof, as in Section 2.4. In other words, in what follows, the observation of $Y(A_1)$ is the first observation in the `bcicov`-dataset, and so on.

3.2 REML estimation of covariance parameters for soil data

In this section we describe the implementation of the estimation method described in Section 2.4. In other words, we wish to perform REML estimation of the covariance parameters of De Wijs plus white noise process, denoted Z in the following, as defined in Section 2.4. We assume a generalised covariance structure on the form

$$K(x-y) = \sigma_0^2 \delta_{x-y} - \sigma_1^2 \log(\|x-y\|), \tag{3.1}$$

where $\sigma_0, \sigma_1 > 0$. Recall that this assumption results in the positive definite covariance matrix

$$\mathbb{C}\mathrm{ov}(Z) = \sigma_0^2 |A_i| \left(I + 11^\top \right) - \sigma_1^2 V, \tag{3.2}$$

where V is a $(n-1) \times (n-1)$ matrix with entrances given by (2.16).
In order to write up the log-likelihood function, as in (2.26), we estimate the matrix V using Monte Carlo methods as in (2.19). In order to do this, we first implement a function which approximates the integral

$$\int_{A_i} \int_{A_j} \log(\|x-y\|) \mathrm{d}x \mathrm{d}y, \tag{3.3}$$

for fixed i and j. Note that the in the implementation of the function, a specific shape, and dimension of the sets A_i is assumed. Specifically, we assume A_i to be a square pillar, with a height of 30.5 cm and a side length of 3.8 cm.

```
MCEst <- function(Ai, Aj, n){
  points <- matrix(NA, n*2, 3)

  for(i in 1:nrow(points)/2){
    points[i,1] <- runif(1, Ai[1]-0.0187, Ai[1]+0.0187)
    points[i,2] <- runif(1, Ai[2]-0.0187, Ai[2]+0.0187)
    points[i,3] <- runif(1,0,0.305)
  }
  for(i in (nrow(points)/2+1):nrow(points)){
    points[i,1] <- runif(1, Aj[1]-0.0187, Aj[1]+0.0187)
    points[i,2] <- runif(1, Aj[2]-0.0187, Aj[2]+0.0187)
    points[i,3] <- runif(1,0,0.305)
  }
  Est <- A^2/n*sum(unlist(lapply(1:n,
           function(x){log(Norm(points[x,] - points[n+x,]))})))
  S2 <- 1/(n-1)*sum(unlist(lapply(1:n,
       function(x){(A^2*log(Norm(points[x,] - points[n+x,])) - Est)^2})))
  return(list(Est = Est, n=n, S2=S2/n))
}
```

Code 3.1: A function approximating (3.3) using Monte Carlo methods for areas A_i and A_j, where centre points of these areas are given as inputs.

The function `MCEst` in Code 3.1 takes three inputs. The first two are the centre points of the areas A_i and A_j around which the points used in the approximation are simulated. The third input determines the number of points to simulate in each area. Note also that `A` is the volume of the index sets (0.00044 m^3). The simulation part of the function consists of two for-loops. The first of these constructs the `n` points in the area A_i, by simulating the x-coordinate, y-coordinate and z-coordinate (depth of sample), uniformly. The second loop does the same for the area A_j. Finally, the approximation is computed as

$$\int_{A_i}\int_{A_j} \log(\|x-y\|)\mathrm{d}x\mathrm{d}y \approx |A_i||A_j|\frac{1}{n}\sum_{m=1}^{n}\log(\|x_m - y_m\|),$$

where x_m is a simulated point in A_i and y_m is a simulated point in A_j. Lastly, we also compute the empirical variance of the estimate, by computing the mean squared difference of the sampled points and the estimate. In order to approximate the integral for all i and j, we use a for-loop as follows.

```
Est.mat <- matrix(NA, 300,300)

for (i in 1:300){
  for (j in 1:i) {
    Est.mat[i,j] <- MCEst(as.numeric(bcicov[i,1:2]),
          as.numeric(bcicov[j,1:2]), n)
  }
}
```

Code 3.2: Monte Carlo approximation of the integral in (3.3) for all i and j.

Using these approximations, we can construct the matrix V, as given in (2.16), using Code 3.3.

```
MC <- Est.mat + t(Est.mat) - diag(Est.mat)
V <- matrix(NA,299,299)
for (i in 2:300) {
  for (j in 2:300) {
    V[i-1,j-1] <- -MC[i,j] + MC[1,j] + MC[i,1] - MC[1,1]
  }
}
```

Code 3.3: Construction of the matrix V using Monte Carlo approximations as in (2.19).

In order to define the log-likelihood as a function of τ as in (2.26), we first define the matrix Φ as in (2.22) in Code 3.4. As previously, `A` is the volume of the areas A_i, for $i = 1, \ldots, n$.

```
Phi <- function(tau){
  A*(eye(299) + 1) + tau*V
}
```

Code 3.4: A function returning the matrix $\Phi(\tau)$ as in (2.22), for a specific value of τ.

Note that due to the way the V matrix is constructed, the second term is added and not subtracted. Using this function, we can implement the estimator of σ_0^2 as a function of the parameter τ. This is done in Code 3.5. Note that `y.tilde` contains the contrasts of data, which is interpreted as a realisation of Z.

```
sigma0.est <- function(tau){
  t(y.tilde)%*%solve(Phi(tau))%*%y.tilde/dim(Phi(tau))[1]
}
```

Code 3.5: A function returning the estimate of σ_0^2 for a given value of τ.

Having implemented the functions `Phi` and `sigma0.est`, we can now implement the log-likelihood as given in (2.26), which is done in Code 3.6.

```
likelihood <- function(tau){
  d <- as.numeric(determinant(Phi(tau), logarithm = TRUE)$modulus)
  l <- -dim(Phi(tau))[1]/2*log(sigma0.est(tau)) - 1/2*d
  return(l)
}
```

Code 3.6: Implementation of the log likelihood function as a function of the parameter τ, as in (2.26).

Now, maximum likelihood estimates of the parameter τ can be obtained. Afterwards, the estimate of σ_0^2 can be obtained by plugging the estimate of τ in the function `sigma0.est` in Code 3.5.

Note that only positive values of τ should be considered, as the likelihood is undefined otherwise.

When using the alternate parametrisation of the GCF presented in a subsection in Section 2.4, a number of challenges arises. Firstly, note that for a generalised random field, Z, with index set as in Section 2.4, which has the GCF

$$K(x,y) = \sigma_0^2 \delta_{x-y} + e^{-\sigma_1^2 \|x-y\|}, \tag{3.4}$$

we must eventually compute the integral

$$\int_{A_i} \int_{A_j} e^{-\sigma_1^2 \|x-y\|} \mathrm{d}x\mathrm{d}y.$$

Since it is not possible separate the parameter σ_1^2 from the integral, the computation is vastly different from the previously discussed case. Firstly, it it not possible to derive the profile log-likelihood is the same way as above. Secondly, the approximations of the integral using Monte Carlo methods is not feasible, as in Code 3.1. This is because the parameter σ_1^2 cannot be separated from the integral, and thus the integral must be approximated for all values of σ_1^2 for which the covariance is to be computed. To mitigate this problem we use a method of approximation which was also discussed in Section 2.4. We use the approximation

$$\int_{A_i} \int_{A_j} e^{-\sigma_1^2 \|x-y\|} \mathrm{d}x\mathrm{d}y \approx |A_i||A_j| e^{-\sigma_1^2 \|x_i - y_j\|},$$

where x_i and y_j are representative points in the sets $|A_i|$ and $|A_j|$, respectively. Using this approximation we can implement the covariance matrix as a function of σ_0^2 and σ_1^2 as in Code 3.7.

```
46 cov <- function(sigma0,sigma1){
47   V <- matrix(NA,299,299)
48   for (i in 2:300) {
49     for (j in 2:300) {
50       if (i==j){
51         V[i-1,j-1] <- 2*A^2*k^sigma1 - 2*A^2*exp.dist[1,i]^sigma1
52       }else{
53         V[i-1,j-1] <- A*exp.dist[i,j]^sigma1
54             - A^2*(exp.dist[1,j]^sigma1 + exp.dist[i,1]^sigma1) + A^2*k^
                 sigma1
55       }
56     }
57   }
58   sigma0*A*(eye(299) + 1) + V
59 }
```

Code 3.7: Function returning the covariance matrix, using the GCF in (3.4) for values of σ_0^2 and σ_1^2 given as input.

Note that in Code 3.7, the notation `exp.dist[i,j]` is the value $e^{-\|x_i - y_j\|}$, where x_i and y_j are representative points in the sets A_i and A_j, respectively.
Using the function defined in Code 3.7, we can define the log-likelihood as function of the parameters σ_0^2 and σ_1^2 as is done in Code 3.8.

```
likelihood <- function(s){
  c <- exp(s)
  co <- cov(c[1],c[2])
  d <- as.numeric(determinant(co, logarithm = TRUE)$modulus)
  l <- -1/2*t(y.tilde)%*%solve(co)%*%y.tilde - 1/2*d
  return(-l)
}
```

Code 3.8: The log-likelihood for a generalised random field with the GCF in (3.4) for given parameters σ_0^2 and σ_1^2.

Note that the function in Code 3.8 returns the negative log-likelihood, and minimising this is equivalent to maximising the log-likelihood. Maximum likelihood estimates can be found using, for example, the function `optim` in R, which minimises multidimensional objective functions.

3.3 Variogram Estimation of the covariance parameters

In this section, we detail the implementation of variogram estimation of covariance parameters. We do this for the GCF model in (3.1), and with the `bcicov`-dataset in mind. As such, some parts may only be applicable to this dataset.

As we have shown in Section 2.6 that for a De Wijs plus white noise process, Z, with index set as in Section 2.4, the variogram can be seen as a function of distance

$$2\gamma(h) = 2\sigma_0^2|A| + 2\sigma_1^2(|A|^2\log(h) - M), \tag{3.5}$$

where $|A|$ is the volume of the sets $|A_i|$ for all i, and M is the approximation of the integral I_{ij} from Section 2.4. This value is approximated before defining any of the following functions. We can define this variogram model as is done in Code 3.9.

```
Var_Model <- function(h, sigma0, sigma1){
  if(h==0){
    m <- M
  }else{
    m <- log(h)*A^2
  }
  return(sigma0*A+sigma1*(m-M))
}
```

Code 3.9: An implementation of the model of the variogram in (3.5).

In Code 3.9, the function takes three inputs. The first is the distance h, as in (3.5). The second and third inputs are the parameter σ_0^2 and σ_1^2. Note that the parameter `sigma0` is actually the squared parameter σ_0^2 in (3.5), and likewise for the parameter `sigma1`. The if-statement is implemented since if $h = 0$ the logarithm $\log(h)$ is undefined, which is problematic. To mitigate this, we implement a temporary variable `m`, which is set to $\log(h)$ for $h > 0$. For $h = 0$ we set this variable to `M`.
In order to estimate the covariance parameters σ_0^2 and σ_1^2 from the variogram model above, we must fit the variogram to the empirical variogram. Therefore we must implement a function which computes the empirical variogram for a given value of h. Recall from Section 2.6 that we can compute the empirical variogram for a given value of h as

$$\hat{\gamma}(h) = \frac{1}{|S(h)|}\sum_{(i,j)\in S(h)}(Y(A_i) - Y(A_j))^2, \tag{3.6}$$

where Y are observations of the underlying random field of Z. Before we are able to implement a function which computes the empirical variogram for a given value h, we first compute a matrix containing the distances $\|x_i - y_j\|$, where x_i and y_j are representative points for A_i and A_j, respective, for all integer values i and j, which both range from 1 to 300 for the `bcicov`-dataset.

```
dist <- matrix(NA, 300,300); temp <- 1
for (i in 1:300) {
  for(j in  1:i){
    if(i!=j){
      dist[i,j] <- Norm(as.numeric(bcicov[i,1:2] - bcicov[j,1:2]))
    }
  }
}
dist[which(is.na(dist))] <- 0
```

Code 3.10: Constructing a matrix where entry (i, j) is the distance between A_i and A_j.

In Code 3.11 we implement a function which computes the empirical variogram for a value of h given as input.

```
variogram <- function(h, y, DistMat){
  sum <- 0
  n <- 0
  for (i in 1:nrow(bcicov)){
    ind <- which(DistMat[i,]==h)
    l <- length(ind)
    if(l!=0){
      for (j in ind){
        sum <- sum + (y[i] - y[j])^2
      }
      n <- n + l
    }
  }
  out <- 1/n*sum
  return(out)
}
```

Code 3.11: A function which, for a given value of h, computes the empirical variogram $\hat{\gamma}(h)$.

The function `variogram` from Code 3.11 takes three inputs. The first is the distance h for which the empirical variogram is to be computed. The second input, `y`, is a $n \times 1$ vector containing the data. Lastly, the third input is a $n \times n$ matrix, `Dist.mat` which contains the distances between the locations of the data in `y`, and this must therefore be computed beforehand.

The main part of the `variogram`-function in Code 3.11 is formed by a for-loop. For each value in the data-vector `y`, it is checked which values are exactly a distance of h away. The indices of these values are stored in the variable `ind`. After this, we check the length of the variable `ind`, to see if any observations lie at the distance of h in question. If there are not any values at this distance, the value in question should not contribute to the sum in (3.6), and we consider the next value in `y`. However, if there are values at the distance of h, we compute the squared difference using these points and count the number of points using the counter `n`. When the for-loop has been run through, the counter `n` attains exactly the value $|S(h)|$, which we then divide the sum by, in order to obtain the empirical variogram. Note that if, for a given distance h, no data is

available the function outputs `NaN`. Note also, that this implementation was checked against the `Variogram`-function from the `nlme`-package, and the two functions give identical results, which we show later.

As noted previously, we implement the variogram specifically for the values of the `bcicov`-dataset. We therefore know the largest possible distance is $h_{max} = 1051.19$. Having computed the distance matrix `dist.mat`, we note that many distances are unique.
We can now implement a function which fits the variogram model to the empirical variogram, by minimizing the Root Mean Square Error (RMSE) between these.

```
36 VarEst <- function(data, obs.points){
37   dist <- matrix(NA, 300,300); temp <- 1
38   for (i in 1:300) {
39     for(j in  1:i){
40       if(i!=j){
41         dist[i,j] <- Norm(as.numeric(obs.points[i,] - obs.points[j,]))
42       }
43     }
44   }
45   dist[which(is.na(dist))] <- 0
46   dist.ordered <- sort(as.numeric(dist))
47   distances <- unique(dist.ordered)
48   var <- unlist(mclapply(distances,
49           function(x){variogram(x, data = data, DistMat = dist)}, mc.
              cores = 4))
50
51   obj.fun <- function(c){
52     s <- exp(c)
53     sigma0 <- s[1]; sigma1 <- s[2]
54     modvar <<- unlist(lapply(distances,
55                  function(x){Var_Model(x, sigma0, sigma1)}))
56     return(sqrt(mse(modvar, var)))
57   }
58   opt <- optim(obj.fun, par=c(10,10))
59   return(list(coef=exp(opt$par), OptVal=opt$value)
60 }
```

Code 3.12: A function fitting the variogram model in Code 3.9 to the empirical variogram, which is computed for the values of h in the vector `distances`.

In Code 3.12, the function `VarEst` starts by constructing the distance matrix as done in Code 3.10. After doing this, the empirical variogram is computed for a data-vector given as input, at the distances in the vector `distances`. After this, we define a function which, for a given value of the parameters σ_0^2 and σ_1^2, computes the RMSE between the variogram model and the empirical variogram. We can minimize this function to obtain estimates of the covariance parameters which minimize the RMSE between the variogram model and the empirical estimate of the variogram. Such minimization can be performed in R, using the function `optim`. Note that the parameters have been transformed, to ensure that the estimates are positive. Note furthermore, that the empirical variogram must be computed before using the function. The

variable `var` is the empirical variogram computed at the values of h in the matrix `distances`. The function outputs the estimate of the parameters and the optimal RMSE value.

Chapter 4

Data Analysis

In this chapter, we analyse the `bcicov`-dataset introduced in Section 3.1, using the implementation described in Chapter 3. In this chapter, we display results from the usage of the implemented functions and state any problem that may occur. Any such problems will be discussed in Chapter 5.

4.1 Maximum likelihood estimation of covariance parameters

In this section, we utilise the implementation of the REML estimation of covariance parameters presented in Section 3.2.

As mentioned in Section 3.1, the dataset contains 19 columns, each containing 300 measurements. Considering the first resource of the dataset, namely Aluminium, it is quickly evident that maximum likelihood estimation for this resource is troublesome. Using optimisation algorithms to maximise the profile log-likelihood, results in the upper bound of the maximisation interval being returned as the maximising point, and changing the interval does not mitigate this. This indicates that the log-likelihood is not bounded. A plot of the profile log-likelihood can be seen in Figure 4.1.

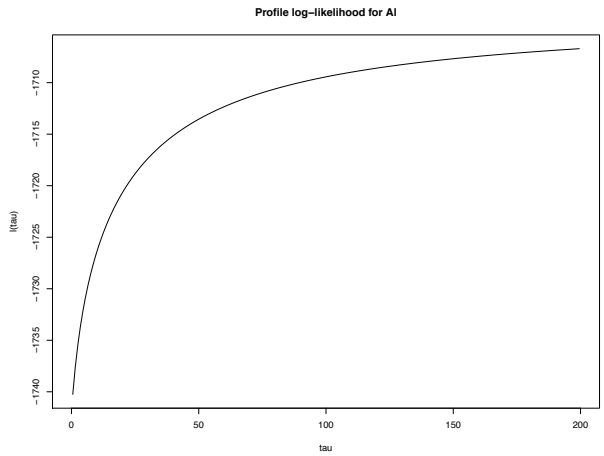


Figure 4.1: Plot of the profile log-likelihood of τ for Aluminium in the bcicov-dataset.

The profile log-likelihood levels out to eventually becomes flat. This results in an estimate of

the parameter which is not unique, i.e. other estimates result in the same profile log-likelihood value. As a result of this, the optimal parameters cannot be identified for Aluminium.

Considering the other resources in the `bcicov`-dataset, it is quickly evident that the flatness of the profile log-likelihood is a reoccurring problem. To investigate this behaviour further, we simulate data using known covariance parameters to see if the behaviour persists. The simulation can be performed using Code 4.1.

```
mu <- rep(0,299)
Cov <- function(sigma, tau){
  sigma*(A*(eye(299) + 1) + tau*V)
}
Sigma <- Cov(1,1)
z <- mvrnorm(1,mu,Sigma)
```

Code 4.1: Simulating a single realisation of a zero mean multivariate normal distribution with covariance parameters $\sigma_0^2 = 1$ and $\sigma_1^2 = 1$.

Recall that the generalised random field, Z, has zero mean and covariance structure as in (2.17). In Code 4.1 we define the covariance matrix as a function of the parameters, and compute the covariance matrix for the specific parameter choices $\sigma_0^2 = 1$ and $\tau = 1$. Lastly, we simulate a single realisation of a multivariate normal distribution using this covariance matrix, and a zero mean vector. The simulated values are denoted `z`. Note that `z` is a vector of contrasts. Using the implementations of Section 3.2, we can compute and plot the profile log-likelihood, as was done with the resources in the `bcicov`-dataset. A plot of the profile log-likelihood can be seen in Figure 4.2.

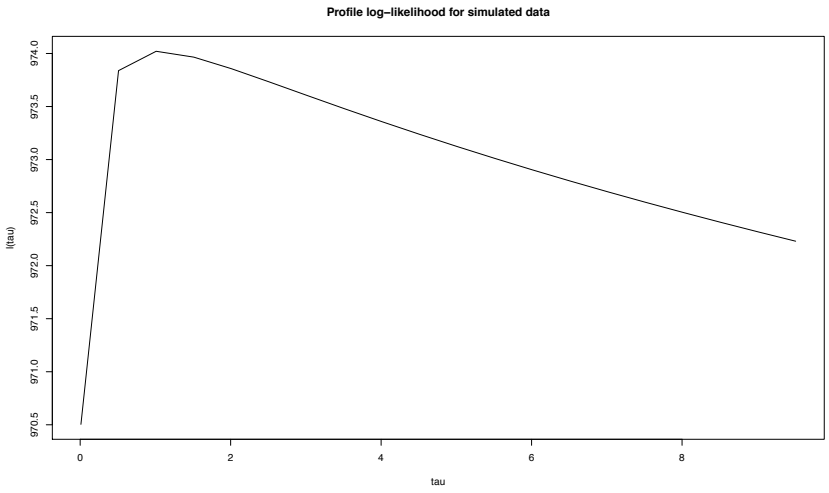


Figure 4.2: A plot of the profile log-likelihood of the parameter τ for simulated data.

We see that the profile log-likelihood is not flat, and it clearly has a maximum point. Using an optimisation function returns the estimate $\hat{\tau} = 1.022$, which is close enough to the true

parameter to be acceptable. Simulating a new realisation and using an optimisation function again, yields the estimate $\hat{\tau} = 0.09$, which is far from the true value. Even using the function for simulated data, seems to yield estimates that vary a lot. In order to asses, the function we simulate multiple realisations and compute the mean of the estimate for each realisation. Suppose we do this 100 times. The mean estimate is then $\hat{\tau}_{\text{mean}} = 14.73$, which is not close. Computing the variance of the estimated parameters, we see that this is 1073.88, which is very high. Setting the true parameter value to $\tau = 10$, and simulating 1000 realisations, yield the mean of the estimates to be $\hat{\tau}_{\text{mean}} = 19.08$, and the variance of the estimate is 998.22.
It seems then that the implemented functions are not accurate for simulated data, and when used on the `bcicov`-dataset, the profile log-likelihood quickly becomes flat. Later we attempt to explain this behaviour.
Using an alternate parametrisation of the GCF, as discussed in Section 2.4 and implemented in Section 3.2, results in log-likelihood functions that briefly increase and quickly become flat, and this parametrisation is therefore not useful.

4.2 Variogram Estimation of Covariance Parameters

In Section 3.3 we implement a function which, given a data-vector as input, estimates the covariance parameters σ_0^2 and σ_1^2 by fitting a variogram model to the empirical variogram. The function is implemented to only utilise the values in the `bcicov`-dataset. We can now use the function `VarEst` from Code 3.12 to estimate the value of the covariance parameters for the various resources in the `bcicov`-dataset.

The estimates for the covariance parameters can be seen in Figure 4.1.

Resource	$\hat{\sigma}_0^2$	$\hat{\sigma}_1^2$	RMSE
Al	266681820.90	298801.55	316903.81
B	223.11	419718.88	2.26
Ca	2516058996.63	45042.86	2734135.78
Cu	14034.77	2364830.78	22.59
Fe	12320096.80	179379.58	14409.63
K	26573856.90	150266.40	27933.27
Mg	98382181.49	357212.35	140446.85
Mn	79652420.83	86217.11	75456.42
Na	13562659.74	94488.13	12197.71
P	14029.85	1.26	16.43
S	141636.66	48189.38	367.20
Zn	0.18	25044968.54	85.38
NH4	238769.46	26785984.69	353.36
NO3	23623.47	25364428.34	91.45
total.N	156322.88	63431266.83	358.23
min.NH4	1.88	265830679.29	711.54
min.NO3	817640.48	38657.08	1087.40
min.N	0.18	580038003.75	1898.16
pH	109.67	119478.19	0.42

Table 4.1: Estimates of covariance parameters, all found using an optimization algorithm with the same initial parameters.

Many of the estimates seem very large. We can asses the validity of estimates visually by comparing the variogram model and the empirical variogram. Consider the second row of Table 4.1, namely the estimates for Boron. In Code 4.2 we compute the empirical variogram and the variogram model using the `VarEst`-function from Section 3.3.

```
B <- VarEst(bcicov$B, bcicov[,1:2])
dist.ordered <- sort(round(as.numeric(dist.mat),0))
distances <- unique(dist.ordered)
mod.plot <- sapply(distances, function(i){Var_Model(i, B$coef[1], B$coef
    [2])})
```

Code 4.2: Computation of the variogram model and empirical variogram using the `VarEst`-function.

A plot of these can be seen in Figure 4.3.

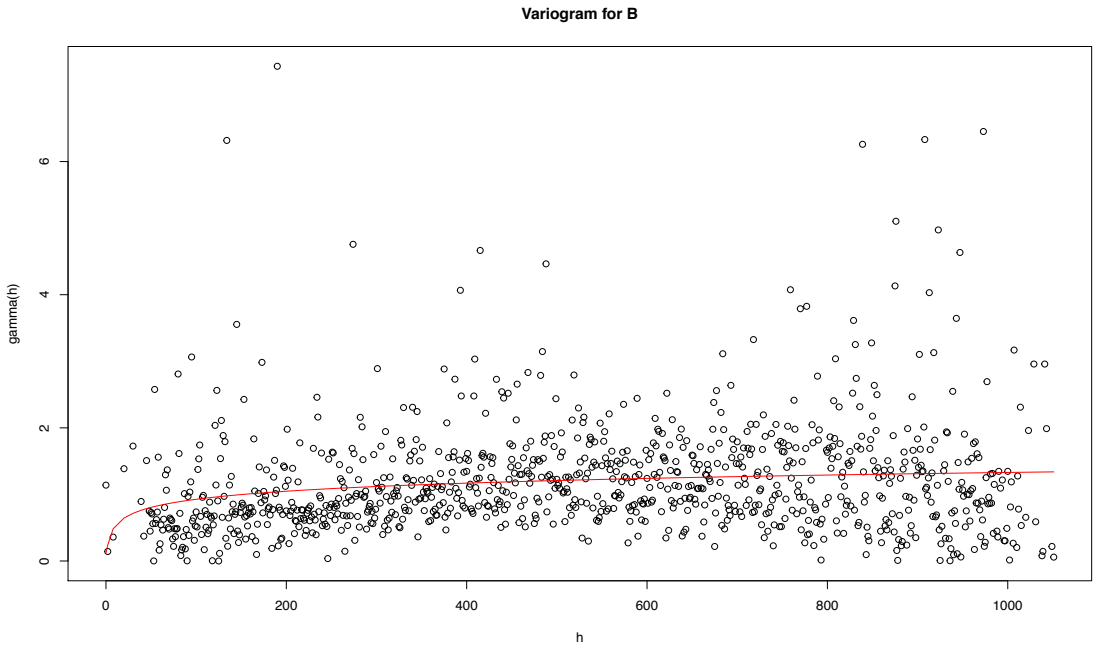


Figure 4.3: Plot of the empirical variogram (circles), and the variogram model (red line) for estimated parameters for Boron.

Note that for clarity purposes, we round the h-values, such that the plots become less cluttered. It can be seen in Figure 4.3, that the variogram model fits the empirical variogram quite well. However, the values of the empirical variogram vary a lot. Notice that some of the large values of the empirical variogram appear to be outliers, as a majority of the values are smaller. Note that the estimated parameters for Boron result in a rather small RMSE of 2.26, which may be due to that the values of the empirical variogram vary a lot. This may also be due to the scale of the data, as this is small for Boron. Consider now the variogram for Aluminium, which can be seen in Figure 4.4.

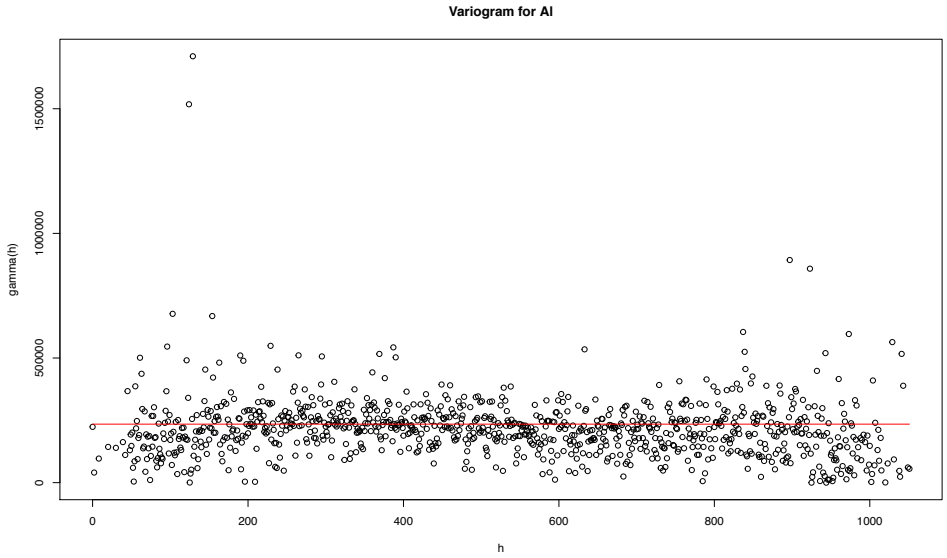


Figure 4.4: Plot of the empirical variogram (circles), and the variogram model (red line) for estimated parameters for Aluminium.

The fitted variogram model for Aluminium arguably fits as well to the empirical variogram, as was the case with Boron. However, note the large outliers. Note also, that the minimised RMSE is 316903.81.

As the `variogram`-function was implemented in Section 3.3, we check the function to make sure it is error free. We do this by computing the empirical variogram for Aluminium, using both the `variogram`-function, and the "native" variogram function, from the `nlme`-package. The plots of both can be seen in Figure 4.5 and Figure 4.6.

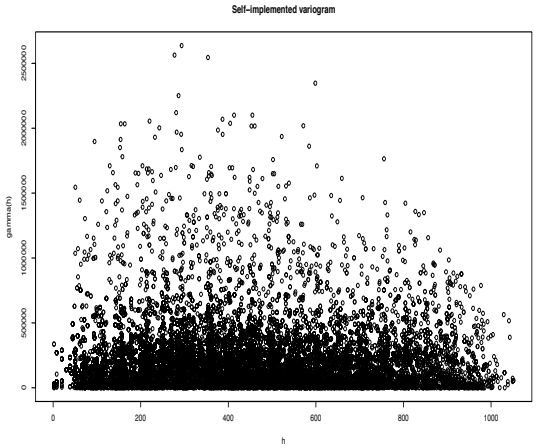


Figure 4.5: Plot of the empirical variogram, computed using the `variogram`-function implemented in Section 3.3.

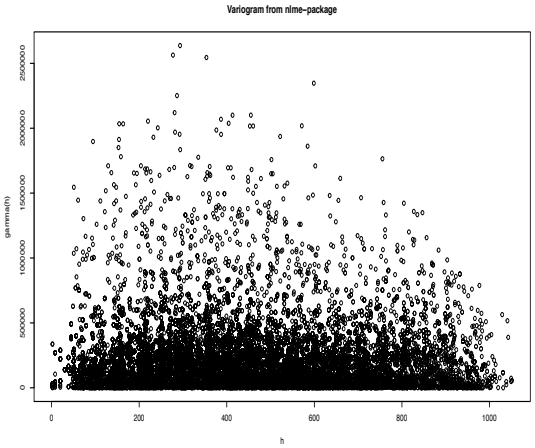


Figure 4.6: Plot of the empirical variogram, computed using the `variogram`-function from the `nlme`-package.

As was done with REML estimation, we check the accuracy of the implementation of variogram estimation using simulated data. Setting the true parameters to $\sigma_0^2 = 1$ and $\sigma_1^2 = 1$, we can obtain multiple estimates and compute the mean of these using Code 4.3.

```
sim.est <- matrix(0,ncol=2, nrow=20)
for (i in 1:20) {
  z <- c(1,mvrnorm(1,mu,Sigma) + 1)
  Z.vario <- VarEst(z, obs.points = bcicov[,1:2])
  sim.est[i,] <- Z.vario$coef
  print(i)
}
mean(sim.est[,1]); mean(sim.est[,2])
```

Code 4.3: Obtaining multiple estimates using simulated data, and computing the mean of these, to asses accuracy of implemented estimation method.

We obtain the mean estimates $\hat{\sigma}_0^2 = 0.0057$ and $\hat{\sigma}_1^2 = 375.04$. We see that the estimate for σ_0^2 is not at all close to the true value. The same is true for $\hat{\sigma}_1^2$. Suppose now, that we set the true values of the covariance parameters to $\sigma_0^2 = 10000$ and $\sigma_1^2 = 10000$. Simulating again, we get that the mean estimates are $\hat{\sigma}_0^2 = 9675.81$ and $\hat{\sigma}_1^2 = 255145.2$. The estimate for σ_0^2 is close to the true value. Computing the empirical standard error of the estimates yields 1996, meaning that the `variogram`-function is relatively accurate in estimating the parameter σ_0^2 when the true value of this parameter is high. As for the parameter σ_1^2, much is left to be desired.

Chapter 5

Discussion and Conclusions

In Chapter 3 we present two implementations of methods, which estimate covariance parameters of the De Wijs plus white noise process, as presented in Section 2.4. Specifically, in Section 3.2 we present an implementation of Restricted Maximum Likelihood estimation(REML), and in Section 3.3 we present variogram estimation.

In the practical application of the implementation of REML estimation, a problem occurs. The profile log-likelihood increases quickly as τ increases, but quickly hereafter it becomes flat. This behaviour is evident for all resources in the `bcicov`-dataset, but the initial increase varies. This may be due to a few reasons. In Clifford and McCullagh [2006], the De Wijs plus white noise model is used to analyse crop yield data. The geometric layout of the data used in the article is different from the geometric layout of the `bcicov`-dataset. In the article, the areas associated with data measurements are large and relatively closely grouped. In the `bcicov`-dataset, this is not the case. The soil data measurements are associated with small sets that are very far apart. It is possible to check whether this contributes to the problem with the REML estimation. To do this, we exclude the data points that do not lie on the grid and pretend that the sets associated with each measurement are a 50×50 square. The areas are now larger, and the distances between areas are zero. We can use a method presented by Clifford [2005] to compute the matrix V in the covariance in (2.17). Doing this, we have a profile log-likelihood as in Figure 5.1.
We can see in Figure 5.1 that this does not mitigate the flatness of the profile log-likelihood. Instead, this seems to have made the problem worse, as the profile log-likelihood increases more quickly than was the case before.
Another cause of trouble is the covariance matrix itself. In Section 2.4, we show that the De Wijs plus white noise process results in a positive definite (and thus invertible) covariance matrix. This is of course also the case in practice, but the determinant of the covariance matrix is so small that, for small values of τ, it is indistinguishable from zero. This could make the inversion of the covariance inaccurate, which would affect the profile log-likelihood.
The covariance matrix consists of the scaled addition of two matrices. The first of which is a matrix of ones, with two on the diagonal, i.e. the diagonal is equal to two times the other elements of the matrix. The second matrix is the matrix V, which consist of differences of approximations of integrals. Looking at the elements of this matrix, it is evident, that the diagonal elements are approximately equal to the other elements times two. The two matrices are thus approximately equal up to scale with a constant. This can mean that the parameters σ_0^2 and σ_1^2, are not

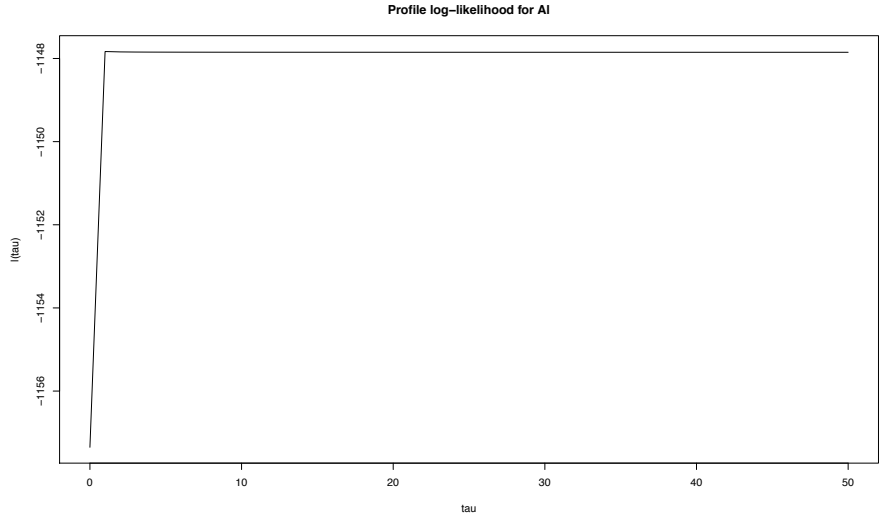


Figure 5.1: A plot of the profile log-likelihood when expanding the index sets of the underlying random field.

separable and that it is possible that only their sum can be estimated. Furthermore, the matrix V consists of values which are approximated using Monte Carlo methods. The code in Code 3.1, makes it possible to compute the empirical variance of the estimates. These are in the order of 10^{-23}, which when compared to the estimates themselves (order of 10^{-9}), indicates that the estimates are relatively accurate.

In regards to the contrasts, we have seen that there many different options. In this project, we have focused on one specific contrast, namely subtracting the first value of the dataset, from the remaining values. This was a choice which was made earlier in the writing process, and it was an arbitrary one at that. Both REML estimation and variogram estimation was derived for this specific contrast. It is doubtful that when looking at the measurements of soil contents, one is only interested in the difference between a measurement and the first measurement. Therefore, the choice to implement the estimation theory for only this contrasts is not optimal. Some other contrasts where attempted used in the implementation, but even though these should result in a positive definite covariance, computationally some were singular. Because of this, the choice was made to continue to focus on a single contrast.

Since the REML estimation of the covariance parameters did not prove feasible, variogram estimation was introduced to estimate the parameters. This approach was then implemented in R, with some parts of the implemented function only working for the dataset introduced in Section 3.1. During the usage of this function, some problems arose. One would expect that the variance of the difference between two variables would increase as the distance increases. This is not always the case, as shown by Figure 4.4, where the variance seems to remain constant, despite a few outliers. It would seem that this is a property of this particular resource in the `bcicov`-dataset. Another thing to note in regards to the estimation of the covariance parameters is that the values of the empirical variogram are quite spread. It is therefore difficult to asses that goodness-of-fit of the model to the empirical variogram. To do this, we use RMSE, but other measures can also be used. In order to check whether the unexpected behaviour of the empirical variogram is a result of an error in the implementation, we compare the self-implemented

function with a native R-function. The two functions return identical values, and we conclude that the `variogram`-function from Section 3.3 performs as desired. In the context of fitting a variogram model to the empirical variogram, we use numerical optimisation through the R function `optim`. Such methods are often inaccurate, and, depending on the form of the objective function, can return different results for different starting values.

Due to time constraints, some aspects of this thesis are merely briefly introduced or outlined. If one had more time, a more comprehensive simulation study could be undertaken, instead of the brief usage in this thesis in order to asses the correctness of the implemented functions. Likewise, as it seems that the data analysed using the implemented functions is not necessarily suited to this kind of analysis, other datasets could be analysed. Another point of interest is the apparent Markov property of the De Wijs process, discussed in Mondal [2015], which was not at all mentioned in this thesis. Lastly, some kind of spatial interpolation (i.e. kriging) using the De Wijs plus white noise process, could be considered.

In conclusion, in Chapter 1, we introduce the notion of generalised function as the foundation on which to introduce generalised stochastic processes. In Chapter 2, we introduce the generalised stochastic process, which allows for more general index set, than the conventional notion of a stochastic process. A special case of the generalised stochastic processes are the generalised random fields, one case of which is of particular interest in this report, namely the De Wijs plus white noise process. For this process (and a modification hereof), we introduced REML estimation of covariance parameters and implemented this approach in the statistical programming language R. Some problems arose while using the implementation. It seems that the data in question is not suited for this process, as simulated data using a known covariance structure resulted in REML estimates, and the example data resulted in flat profile log-likelihood. Using the variogram estimation method from Section 2.6, similar yields unexpected results and behaviour of the empirical variogram for some resources, which may support this conclusion. Additionally, inaccuracy in the estimation of the covariance parameters using variogram may be due to the numerical methods used for fitting a variogram model to the empirical variogram.

Bibliography

Herbert Amann and Joachim Escher. *Analysis II*. Birkhauser, 2008.

Christian Berg and Tage Gutmann Madsen. Mål- og Integrationsteori. Unpublished Lecture Notes.

Julian Besag and Debashis Mondal. First-order intrinsic autoregressions and the de wijs process. *Biometrika*, 92(4):909–920, 2005. ISSN 00063444. URL http://www.jstor.org/stable/20441244.

David Clifford. Computation of spatial covariance matrices. *Journal of Computational and Graphical Statistics*, 14(1):155–167, 2005.

David Clifford and Peter McCullagh. Evidence for conformal invariance of crop yields. *Proceedings of the Royal society*, 2006.

D. A. Dawson. Generalized Stochastic Integrals and Equations. *Transactions of the American Mathematical Society*, 147:473–506, 1970.

Kyle Harms, James W. Dalling, Joseph B. Yavitt, and Robert Stallard. Forest dynamics plot soil sampling protocols, September 2004.

Ib Thorsgaard Jensen and Nicholas Fitzhugh. Intrinsic Random Fields of Order k - Theory and Implementation. Semesterproject on the ninth semester, 2018.

G. Matheron. Traité de Géostatistique Appliquée. *Editions Technip*, 1962.

G. Matheron. *The Theory of Regionalized Variables and Its Applications*. Centre de Morphologie Mathématique Fontainebleau: Les cahiers du Centre de Morphologie Mathématique de Fontainebleau. École national supérieure des mines, 1971. URL https://books.google.dk/books?id=TGhGAAAAYAAJ.

L. Mattner. Strict definiteness of integrals via complete monotonicity of derivatives. *Transactions of the American Mathematical Society*, 349(8):3321–3342, 1997. ISSN 00029947.

Peter McCullagh. What is a statistical model? *Ann. Statist.*, 30(5):1225–1310, 10 2002. doi: 10.1214/aos/1035844977. URL https://doi.org/10.1214/aos/1035844977.

Peter McCullagh. How Oats and Beans and Barley Grow. Abstract of talk given at Statistics Department of Oxford Univeristy, October 2003. URL http://www.stats.ox.ac.uk/~bithell/seminars/mccullagh.abstract.pdf.

D. Mondal. Applying Dynkin's isomorphism: An alternative approach to understand the Markov property of the de Wijs process. *Bernoulli*, 21(3):1289–1303, aug 2015. doi: 10.3150/13-bej541.

Ebbe Thue Poulsen. *Funktioner af en og flere variable*. Gad, third edition, 2015.

S. Rongfeng. Lecture 5. Lecture notes on the convergence of random variables.

S. Schäffler. *Generalized Stochastic Processes*. Birkhauser, 2018. ISBN 978-3-319-78767-1.

Rasmus Waagepetersen. Maximum Likelihood Estimation for Linear Mixed Models. Lecture slides, February 2019. URL http://people.math.aau.dk/~rw/Undervisning/MM_BI/Handouts/handout2-2x2.pdf.