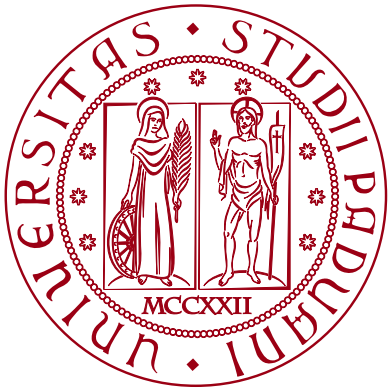


Università degli Studi di Padova

Dipartimento di Matematica "Tullio Levi-Civita"

Corso di Laurea Magistrale in Matematica

Application of kernel-based Bayesian optimization in 3D CZT SPECT reconstructions

Relatore:
Dott. Francesco Marchetti

Candidato:
Luca Pastrello
2058293

19 Luglio 2024

Alla mia famiglia

Contents

Introduction

Someone could wonder about the connection between mathematics and medicine.

At first glance these two fields might seem unrelated, but the intersection of medicine, physics and mathematics has given rise to groundbreaking advancements in medical imaging technologies such as, for example, Single Photon Emission Computed Tomography (SPECT). SPECT imaging is a powerful diagnostic tool in nuclear medicine, providing detailed 3D images of internal organs. The quality of these images is crucial for accurate diagnosis and treatment planning.

In the Department of Nuclear Medicine of the University of Padova, the StarGuide detector is an advanced SPECT/CT imaging device. However, the image reconstructions provided by the StarGuide are heavily dependent on the chosen configuration of various parameters. Therefore, in order to obtain high-quality reconstructions and fully exploit its technology and capabilities, the challenge consists of carefully tuning such parameters.

From a mathematical viewpoint, to deal with this problem, the first step consists in defining a suitable "quality assessment" target function, which should also be established in agreement with the expert physician and may assume various mathematical properties. The second step then consists of setting an effective strategy to maximize the chosen target function. For this purpose, in this thesis we will consider kernel-based Bayesian Optimization (BO). BO is a statistical method for maximizing possible non-concave functions, which relies on surrogate models that approximate the target function by means of positive definite kernels. Furthermore, BO is particularly useful for problems where the objective function is expensive to evaluate or is known only for a restricted set of data, as in the StarGuide image reconstruction problem.

For these reasons, the findings and results of this research work have the potential to significantly enhance the diagnostic capabilities of SPECT imaging with a fine-tuned StarGuide, leading to more accurate diagnoses and better patient outcomes in clinical settings.

The structure of the thesis unfolds through interconnected chapters. Chapter 1 lays the foundational mathematical preliminaries, essential for understanding Reproducing Kernel Hilbert Spaces (RKHSs). Therefore,

Chapter 2 delves into RKHSs, providing theoretical results that are linked to the properties and capabilities of the kernel-based BO algorithm introduced in Chapter 3, where details on its implementation and effectiveness are presented and also exposed via some numerical tests.

Chapter 4 examines various image quality metrics that are crucial in constructing the target function that will be used in our optimization task. Chapter 5 shifts focus to nuclear medicine and SPECT imaging, describing concepts which are essential in order to understand our application. Finally, Chapter 6 applies BO to find the best parameter configuration for the StarGuide, showing the obtained results.

Chapter 1

Mathematical preliminaries and notations

The following definitions are taken from [3].

In order to understand the mathematical objects that are presented in this work we recall some definitions and notions that we have to keep in mind for a better comprehension.

1.1 Brief review of normed vector spaces

The definitions of this section are fundamental to have a clear reading of the Chapter 2.

Definition 1.1.1 (**Vector space**)**.** A *vector space* consists of a set X (elements of X are called *vectors*), a field $\mathbb{F}$ (elements of $\mathbb{F}$ are called *scalars*), and two operations.

1. An operation called *vector addition* that takes two vectors $\boldsymbol{v}, \boldsymbol{w} \in X$, and produces a third vector, written $\boldsymbol{v} + \boldsymbol{w} \in X$.

2. An operation called *scalar multiplication* that takes a scalar $c \in \mathbb{F}$ and a vector $\boldsymbol{v} \in X$, and produces a new vector, written $c\boldsymbol{v} \in X$.

Definition 1.1.2 (**Normed vector space**)**.** Let X be a $\mathbb{F}$-vector space. A *norm* on X is a function $||\cdot|| : X \to \mathbb{R}$ that satisfies the following properties:

1. $||\boldsymbol{x}|| \geq 0$, $||\boldsymbol{x}|| = 0$ if and only if $\boldsymbol{x} = 0$;

2. $||\lambda \boldsymbol{x}|| = |\lambda| ||\boldsymbol{x}||$, for all $\lambda \in \mathbb{F}$ and $\boldsymbol{x} \in X$.

3. $||\boldsymbol{x} + \boldsymbol{y}|| \leq ||\boldsymbol{x}|| + ||\boldsymbol{y}||$, for all $\boldsymbol{x}, \boldsymbol{y} \in X$

A *normed vector space* is a $\mathbb{F}$-vector space provided with a norm.

Given a normed vector space $(X, ||\cdot||)$ and $\boldsymbol{x}, \boldsymbol{y} \in X$, the function defined as

$$\begin{aligned} d : X \times X &\to [0, +\infty) \\ (\boldsymbol{x}, \boldsymbol{y}) &\mapsto d(\boldsymbol{x}, \boldsymbol{y}) := ||\boldsymbol{x} - \boldsymbol{y}|| \end{aligned}$$

satisfies the following properties

1. $d(\boldsymbol{x}, \boldsymbol{x}) = 0$ for all $\boldsymbol{x} \in X$, and if $d(\boldsymbol{x}, \boldsymbol{y}) = 0$ then $\boldsymbol{x} = \boldsymbol{y}$;
2. $d(\boldsymbol{x}, \boldsymbol{y}) = d(\boldsymbol{y}, \boldsymbol{x})$ for all $x, y \in X$;
3. $d(\boldsymbol{x}, \boldsymbol{z}) \leq d(\boldsymbol{x}, \boldsymbol{y}) + d(\boldsymbol{y}, \boldsymbol{z})$ for all $\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{z} \in X$;

such a function is called *distance*. The couple (X, d) is called *metric space*.

Definition 1.1.3 (**Cauchy sequence**)**.** Let (X, d) be a metric space. A sequence $(\boldsymbol{x}_j)_{j \in \mathbb{N}}$ of points of X is said *Cauchy sequence* if, for all $\varepsilon > 0$, $\exists n_\varepsilon \in \mathbb{N}$ such that for every couple of indices $n, m \geq n_\varepsilon$ we have $d(\boldsymbol{x}_n, \boldsymbol{x}_m) \leq \varepsilon$.

Definition 1.1.4 (**Complete metric space**)**.** A metric space (X, d) is called *complete* if every Cauchy sequence of points in X has a limit that is also in X.

Definition 1.1.5 (**Dense subset of a metric space**)**.** A subset A of a topological space X is said to be a *dense subset* of X if the smallest closed subset of X containing A is X itself.

In the case of metric spaces, when we have that the topology of X is induced by a metric, we have that A is a dense subset of X , if the closure of A, so the union of A with the set of all the limits of the sequences of A:

$$\bar{A} = \{ \lim_{n \to \infty} \boldsymbol{a}_n : \boldsymbol{a}_n \in A \quad \forall n \in \mathbb{N} \},$$

is such that $\bar{A} = X$.

Definition 1.1.6 (**Pre-Hilbert space**)**.** A *Pre-Hilbert* space is a couple $(H, \langle \cdot, \cdot \rangle)$ where H is a real or complex vector space and $\langle \cdot, \cdot \rangle$ is an inner product.

Definition 1.1.7 (**Hilbert space**)**.** A *Hilbert space* is a real or complex vector space provided with an inner product that is also a complete metric space with respect to the distance function induced by the inner product.

1.2 Brief overview on the Fourier transform

Now we recall the definition of *Fourier transform* which is a mathematical tool that will be used in Chapter 4.[12]

Definition 1.2.1 (**Fourier transform**). Let $f \in L^1(\mathbb{R}^n)$, the *n-dimensional Fourier transform* of f is

$$(Ff)(\boldsymbol{u}) = \int_{\mathbb{R}^n} f(\boldsymbol{x}) e^{-i2\pi\langle \boldsymbol{x},\boldsymbol{u}\rangle} d\boldsymbol{x} \tag{1.1}$$

Definition 1.2.2 (**Inverse Fourier transform**). Let $f \in L^1(\mathbb{R}^n)$, the *n-dimensional inverse Fourier transform of* f is

$$(F^{-1}f)(\boldsymbol{x}) = \int_{\mathbb{R}^n} f(\boldsymbol{u}) e^{i2\pi\langle \boldsymbol{x},\boldsymbol{u}\rangle} d\boldsymbol{u}. \tag{1.2}$$

Now we show an example of a computation of a Fourier Transform of a particular function.

Example 1.2.1. Let $\varphi(\boldsymbol{x}) = e^{-\pi||\boldsymbol{x}||^2}$ which takes value on $\mathbb{R}$. We want to compute $(F\varphi)$.

$$\begin{aligned}
(F\varphi)(\boldsymbol{y}) &= \int_{\mathbb{R}^n} e^{-\pi||\boldsymbol{u}||^2} e^{-i2\pi\langle \boldsymbol{u},\boldsymbol{y}\rangle} d\boldsymbol{u} \\
&= \int_{\mathbb{R}^n} e^{-\pi(||\boldsymbol{u}||^2 + i2\pi\langle \boldsymbol{u},\boldsymbol{y}\rangle - ||\boldsymbol{y}||^2 + ||\boldsymbol{y}||^2)} d\boldsymbol{u} \\
&= e^{-\pi||\boldsymbol{y}||^2} \int_{\mathbb{R}^n} e^{-\pi||\boldsymbol{u}+i\boldsymbol{y}||^2} d\boldsymbol{u} \\
&= e^{-\pi||\boldsymbol{y}||^2} = \varphi(\boldsymbol{y}),
\end{aligned}$$

where we have used $\int_{\mathbb{R}^n} e^{-\pi||\boldsymbol{u}+i\boldsymbol{y}||^2} d\boldsymbol{u} = 1$. This fact comes from a change of variables in the integral, because $\boldsymbol{y}$ is a fixed vector so

$$\int_{\mathbb{R}^n} e^{-\pi||\boldsymbol{u}+i\boldsymbol{y}||^2} d\boldsymbol{u} = \int_{\mathbb{R}^n} e^{-\pi||\boldsymbol{u}||^2} d\boldsymbol{u},$$

which can be split using Fubini-Tonelli's Theorem

$$\int_{\mathbb{R}^n} e^{-\pi||\boldsymbol{u}||^2} d\boldsymbol{u} = \int_{\mathbb{R}} e^{-\pi u_1^2} du_1 \cdots \int_{\mathbb{R}} e^{-\pi u_n^2} du_n,$$

and $\int_{\mathbb{R}} e^{-\pi x^2} dx = 1$.

From this example we have proved that

$$(F\varphi)(\boldsymbol{y}) = \varphi(\boldsymbol{y}). \tag{1.3}$$

This result is crucial for the proof of the following theorem, which explains the reason why (1.2) is called inverse Fourier transform.

Theorem 1.2.1 (**Fourier inversion theorem**). *Let $f \in L^1(\mathbb{R}^n)$. Then for all $\boldsymbol{x} \in \mathbb{R}^n$*

$$(F^{-1}(Ff))(\boldsymbol{x}) = f(\boldsymbol{x}).$$

Proof. We notice that $Ff \in L^1(\mathbb{R}^n)$ because $f \in L^1(\mathbb{R}^n)$, so by the Dominated Convergence Theorem

$$\int_{\mathbb{R}^n} e^{i2\pi\langle \boldsymbol{x},\boldsymbol{u}\rangle}(Ff)(\boldsymbol{u})d\boldsymbol{u} = \lim_{\varepsilon\to 0}\int_{\mathbb{R}^n} e^{-\pi\varepsilon^2||\boldsymbol{u}||^2+i2\pi\langle \boldsymbol{x},\boldsymbol{u}\rangle}(Ff)(\boldsymbol{u})d\boldsymbol{u}.$$

We define $g_{\boldsymbol{x}}(\boldsymbol{u}) = e^{-\pi\varepsilon^2||\boldsymbol{u}||^2+2i\pi\langle \boldsymbol{x},\boldsymbol{u}\rangle}$ and apply the Fourier transform on it, we obtain

$$\begin{aligned}
(Fg_x)(\boldsymbol{y}) &= \int_{\mathbb{R}^n} g_{\boldsymbol{x}}(\boldsymbol{u})e^{-i2\pi\langle \boldsymbol{y},\boldsymbol{u}\rangle}d\boldsymbol{u} \\
&= \int_{\mathbb{R}^n} e^{-\pi\varepsilon^2||\boldsymbol{u}||^2+i2\pi\langle \boldsymbol{x},\boldsymbol{u}\rangle-i2\pi\langle \boldsymbol{y},\boldsymbol{u}\rangle}d\boldsymbol{u} \\
&= \int_{\mathbb{R}^n} e^{i2\pi\langle (\boldsymbol{x}-\boldsymbol{y}),\boldsymbol{u}\rangle-\pi\varepsilon^2||\boldsymbol{u}||^2}d\boldsymbol{u} \\
&= \int_{\mathbb{R}^n} e^{-i2\pi\langle (\boldsymbol{y}-\boldsymbol{x}),\boldsymbol{u}\rangle}\psi(\boldsymbol{u})d\boldsymbol{u} \\
&= (F\psi)(\boldsymbol{y}-\boldsymbol{x}),
\end{aligned}$$

where $\psi(\boldsymbol{u}) = e^{-\pi\varepsilon^2||\boldsymbol{u}||^2}$. Now, if $\varphi = e^{-\pi||\boldsymbol{v}||^2}$, we have that $\psi(\boldsymbol{u}) = \varphi(\varepsilon\boldsymbol{u})$. In the previous integral we apply the substitution $\boldsymbol{u} = \frac{\boldsymbol{v}}{\varepsilon}$ so for a well known theorem we have

$$\begin{aligned}
\int_{\mathbb{R}^n} e^{-i2\pi\langle (\boldsymbol{y}-\boldsymbol{x}),\boldsymbol{u}\rangle}\psi(\boldsymbol{u})d\boldsymbol{u} &= \frac{1}{\varepsilon^n}\int_{\mathbb{R}^n} e^{-i2\pi\left\langle \frac{(\boldsymbol{y}-\boldsymbol{x})}{\varepsilon},\boldsymbol{v}\right\rangle}\psi\left(\frac{\boldsymbol{v}}{\varepsilon}\right)d\boldsymbol{v} \\
&= \frac{1}{\varepsilon^n}\int_{\mathbb{R}^n} e^{-i2\pi\left\langle \frac{(\boldsymbol{y}-\boldsymbol{x})}{\epsilon},\boldsymbol{v}\right\rangle}\varphi(\boldsymbol{v})d\boldsymbol{v} \\
&= \frac{1}{\varepsilon^n}(F\varphi)\left(\frac{\boldsymbol{y}-\boldsymbol{x}}{\varepsilon}\right) \\
&= \frac{1}{\varepsilon^n}\varphi\left(\frac{\boldsymbol{y}-\boldsymbol{x}}{\varepsilon}\right) = \varphi_\varepsilon(\boldsymbol{y}-\boldsymbol{x}),
\end{aligned}$$

where we have used the fact $(F\varphi)(\boldsymbol{y}) = \varphi(\boldsymbol{y})$ and the definition $\frac{1}{\varepsilon^n}\varphi\left(\frac{\boldsymbol{x}}{\varepsilon}\right) = \varphi_\varepsilon(\boldsymbol{x})$. The chain of equalities leads to $(Fg_{\boldsymbol{x}})(\boldsymbol{y}) = \varphi_\varepsilon(\boldsymbol{y}-\boldsymbol{x})$. We consider the integral of the beginning and by Fubini-Tonelli's Theorem we attain

$$\begin{aligned}
\lim_{\varepsilon\to 0}\int_{\mathbb{R}^n} e^{-\pi\varepsilon^2||\boldsymbol{u}||^2+i2\pi\langle \boldsymbol{x},\boldsymbol{u}\rangle}(Ff)(\boldsymbol{u})d\boldsymbol{u} &= \lim_{\varepsilon\to 0}\int_{\mathbb{R}^n} g_{\boldsymbol{x}}(\boldsymbol{u})(Ff)(\boldsymbol{u})d\boldsymbol{u} \\
&= \lim_{\varepsilon\to 0}\int_{\mathbb{R}^n}\int_{\mathbb{R}^n} g_x(\boldsymbol{u})e^{-i2\pi\langle \boldsymbol{y},\boldsymbol{u}\rangle}f(\boldsymbol{y})d\boldsymbol{y}d\boldsymbol{u} \\
&= \lim_{\varepsilon\to 0}\int_{\mathbb{R}^n} (Fg_{\boldsymbol{x}})(\boldsymbol{y})f(\boldsymbol{y})d\boldsymbol{y} \\
&= \lim_{\varepsilon\to 0}\int_{\mathbb{R}^n} \varphi_\varepsilon(\boldsymbol{y}-\boldsymbol{x})f(\boldsymbol{y})d\boldsymbol{y} = (\varphi_\varepsilon * f)(\boldsymbol{x}),
\end{aligned}$$

which is the convolution of f with an approximate identity. But since $f \in L^1(\mathbb{R}^n)$, we have that

$$\lim_{\varepsilon \to 0} (\varphi_\varepsilon * f)(\boldsymbol{x}) = f(\boldsymbol{x}).$$

Putting everything together we have shown the equality

$$\int_{\mathbb{R}^n} e^{i2\pi\langle \boldsymbol{x}, \boldsymbol{u}\rangle} (Ff)(\boldsymbol{u}) d\boldsymbol{u} = f(\boldsymbol{x}).$$

□

Observation 1.2.1. Now we consider the following set of functions:

$$E_n = \text{span}\{e^{i2\pi\langle \boldsymbol{x}, \boldsymbol{u}\rangle} : \boldsymbol{u} \in \mathbb{R}^n\}.$$

From the Fourier inversion theorem, we obtain that E_n is dense in the set of integrable functions because, as its consequence, we can write all the functions which belong to $L^1(\mathbb{R}^n)$ as superposition of these exponentials. We can also see that the complex exponentials are orthogonal functions in $L^2(\mathbb{R}^n)$.

Let $\boldsymbol{u}, \boldsymbol{v} \in \mathbb{R}^n$, we have:

$$\int_{\mathbb{R}^n} e^{i2\pi\langle \boldsymbol{x}, \boldsymbol{u}\rangle} e^{-i2\pi\langle \boldsymbol{x}, \boldsymbol{v}\rangle} d\boldsymbol{x} = \int_{\mathbb{R}^n} e^{i2\pi\langle \boldsymbol{x}, \boldsymbol{u}-\boldsymbol{v}\rangle} d\boldsymbol{x}.$$

By Fubini-Tonelli's Theorem we get

$$\int_{\mathbb{R}^n} e^{i2\pi\langle \boldsymbol{x}, \boldsymbol{u}-\boldsymbol{v}\rangle} d\boldsymbol{x} = \int_{\mathbb{R}} e^{i2\pi x_1(u_1-v_1)} dx_1 \cdots \int_{\mathbb{R}} e^{i2\pi x_1(u_n-v_n)} dx_n.$$

Now we can consider the integrals independently:

$$\int_{\mathbb{R}} e^{i2\pi kx} dx = \int_{\mathbb{R}} \cos(2\pi kx) + i \sin(2\pi kx) dx$$

which is ∞ when $k = 0$ and 0 when $k \neq 0$.

So the previous chain of equality is ∞ if $u_i = v_i \ \forall i$, 0 if $\exists i : u_i \neq v_i$.

This proves the orthogonality and ensures that the basis functions do not overlap in their contribution to the functions decomposition.

1.3 Fourier transform in image processing

Now we can attain the contribution of the Fourier transform in digital image processing: the Fourier transform and its inverse provide the formal mechanism for decomposing a function into its frequency components and reconstructing it.

This is an immediate result of the previous theorem and observations. In order to see it we follow the steps:

1. **Existence of Fourier transform.** Let f the function which describes a certain image.

   As we have said in the introduction, an image can be seen as a function, in particular as a simple function $f : \Omega \to \mathbb{R}$, where Ω is a compact subset of $\mathbb{R}^2$ and represents the grid of pixels on which the function takes values between 0 and 255.

   Since f is a function on a compact set, it is surely contained in $L^1(\Omega)$. We can extend the function to $\mathbb{R}^2$, assuming that is null outside Ω, obviously we have obtained a function which still belongs to $L^1(\mathbb{R}^2)$.

   Since $f \in L^1(\mathbb{R}^2)$, its Fourier transform exists and we can write

$$(Ff)(u,v) = \int_{\mathbb{R}^2} f(x,y)e^{-i2\pi(ux+vy)}dxdy.$$

2. **Representation.** Now by Theorem 1.2.1 we can express $f(x,y)$ in terms of its frequency components:

$$f(x,y) = \int_{\mathbb{R}^2} (Ff)(u,v)e^{i(2\pi(ux+vy))}dudv.$$

3. **Uniqueness of the representation.** We suppose that f can be written with two different combinations of the basis functions E:

$$\begin{aligned} 0 &= f(x,y) - f(x,y) \\ &= \int_{\mathbb{R}^2} (Ff)_1(u,v)e^{i(2\pi(ux+vy))}dudv - \int_{\mathbb{R}^2} (Ff)_2(u,v)e^{i(2\pi(ux+vy))}dudv \\ &= \int_{\mathbb{R}^2} ((Ff)_1(u,v) - (Ff)_2(u,v))e^{(2\pi(ux+vy))}dudv. \end{aligned}$$

   Then, for the orthogonality of the basis functions, the coefficients should be zero, so $(Ff)_1(u,v) = (Ff)_2(u,v)$ and this leads to the uniqueness of the representation.

Finally, if f is the function which describes a certain image we can write it as a unique combination of elements of E_2.

Now we have completed the set up in order to understand the role of Fourier transform in process imaging.

After the previous reasoning, we have learnt that we can write a signal as a combination of sinusoids each of them characterised by a certain frequency (u,v) and multiplied by the respective Fourier transform $(Ff)(u,v)$.

The Fourier transform is generally a complex function so it can be expressed as:

$$(Ff)(u,v) = |(Ff)(u,v)|e^{i\phi(u,v)}.$$

The two factors which appears in this product are essential to the complete characterization of the reconstructed signal.

$|(Ff)(u,v)|$ is called *Fourier magnitude spectrum* and represents the amplitude of the frequency components. It describes how much of each frequency is present in the image.[10]

$\phi(u,v)$ is called *Fourier phase spectrum* shows the position or phase shift of the frequency components because, if we look at the (1.2), it multiplies the sinusoids.[10]

Now we show the reason why the Fourier phase spectrum of an image contains crucial information about image geometry, in particular its contour.

In order to see this we show that a slight modification of the phase of the elements of the superposition (1.2) can lead to a "translation" of certain parts of the image.

Proposition 1.3.1. ***(Translation in the Spatial Domain by Phase Multiplication in the Frequency Domain)*** *Let $f : \Omega \to \mathbb{R}$ be a function on a compact set $\Omega \subset \mathbb{R}^n$ and $(Ff)(\boldsymbol{u})$ its Fourier transform, then multiplying the Fourier transform of $f(\boldsymbol{x})$ by a complex exponential $e^{-i2\pi\langle \boldsymbol{u},\boldsymbol{x}_0\rangle}$ results in a translation of the original function $f(\boldsymbol{x})$ by $\boldsymbol{x}_0$ in the spatial domain.*

Proof. The Fourier transform of f is (1.1). Then, through the Theorem 1.2.1 we can express f in terms of its frequency components. We attain

$$f(\boldsymbol{x}) = \int_{\mathbb{R}^n} F(\boldsymbol{u}) e^{i2\pi\langle \boldsymbol{x},\boldsymbol{u}\rangle} d\boldsymbol{u}.$$

Now we multiply by $e^{-i2\pi\langle \boldsymbol{x}_0,\boldsymbol{u}\rangle}$ the frequency components and we obtain

$$\tilde{F}(\boldsymbol{u}) = F(\boldsymbol{u}) e^{-i2\pi\langle \boldsymbol{x}_0,\boldsymbol{u}\rangle}.$$

Therefore, we use (1.2),

$$\begin{aligned}(F^{-1}\tilde{F})(\boldsymbol{x}) &= \int_{\mathbb{R}^n} \tilde{F}(\boldsymbol{u}) e^{i2\pi\langle \boldsymbol{x},\boldsymbol{u}\rangle} d\boldsymbol{u} \\ &= \int_{\mathbb{R}^n} F(\boldsymbol{u}) e^{-i2\pi\langle \boldsymbol{x}_0,\boldsymbol{u}\rangle} e^{i2\pi\langle \boldsymbol{x},\boldsymbol{u}\rangle} \\ &= \int_{\mathbb{R}^n} F(\boldsymbol{u}) e^{i2\pi\langle \boldsymbol{u},(\boldsymbol{x}-\boldsymbol{x}_0)\rangle} d\boldsymbol{u}.\end{aligned}$$

Now we can recognize that this is just the multidimensional inverse Fourier Transform of $F(\boldsymbol{u})$ evaluated at $(\boldsymbol{x} - \boldsymbol{x}_0)$, so

$$\tilde{f}(\boldsymbol{x}) = f(\boldsymbol{x} - \boldsymbol{x}_0).$$

□

We apply the previous proposition to the bidimensional case. Let f be a $2D$ signal, we express it in terms of its frequency components through the inversion Fourier theorem as we have done before

$$f(x,y) = \int_{\mathbb{R}^2} F(u,v) e^{i2\pi(ux+vy)} du dv.$$

We apply a phase shift to the frequency components, we obtain the new coefficients:

$$\tilde{F}(u,v) = F(u,v) e^{i\phi(u,v)}.$$

We assume that $\phi(u,v) = -2\pi(ux_0 + vy_0)$, then

$$\tilde{F}(u,v) = F(u,v) e^{-i2\pi(ux_0+vy_0)},$$

applying the inverse $2D$ Fourier Transform to the phase-shifted frequency domain representation:

$$\tilde{f}(x,y) = \int_{\mathbb{R}^2} F(u,v) e^{i2\pi(u(x-x_0)+v(y-y_0))} du dv.$$

Now we can recognize that this is just the inverse 2D Fourier Transform of $F(u,v)$ evaluated at $(x - x_0, y - y_0)$, so

$$\tilde{f}(x,y) = f(x - x_0, y - y_0).$$

The conclusion is that varying the phase term we have a uniform translation in the spatial domain.

This leads to a problem: when there is a phase shift, the image will be translated beyond the boundaries of the image frame and so it will disappear from the visible area if we are considering a finite image or discrete domain.

In general, in order to solve this problem, if $f : \Omega \to \mathbb{R}$, where Ω is a compact rectangular subset of $\mathbb{R}^2$, we can perform the Ω-periodization of $\boldsymbol{u}$.

In this way, pixels that move beyond an edge are wrapped around to the opposite edge, and we have no loss of information.

Observation 1.3.1 (**Increment by a Gaussian random field**)**.** We can understand better the nature of the information contained in the Fourier phase spectrum multiplying it with a different function. In particular, we introduce a random vector field that modifies the value of the basis functions.

We define the new coefficients

$$(\tilde{F}f)(\boldsymbol{u}) = (Ff)(\boldsymbol{u}) e^{i2\pi\varepsilon S(\boldsymbol{u})},$$

where $\epsilon > 0$ and S is a Gaussian vector field: $S \sim N(\mu, \sigma^2)$.

The Gaussian vector field tries to describe a noise. The multiplication by ε modifies the intensity. If $\varepsilon \approx 0$, so if we try to describe a small factor of disturbance, we have

$$\begin{aligned}(\tilde{F}f)(\boldsymbol{u}) &= (Ff)(\boldsymbol{u})(\cos(2\pi\varepsilon S(\boldsymbol{u})) + i\sin(2\pi\varepsilon S(\boldsymbol{u}))) \\ &\approx (Ff)(\boldsymbol{u})(1 + i\varepsilon 2\pi S(\boldsymbol{u})) \\ &\approx (Ff)(\boldsymbol{u}) + i\varepsilon 2\pi (Ff)(\boldsymbol{u})S(\boldsymbol{u})\end{aligned}$$

Now we apply to $(\tilde{F}f)$ the inverse Fourier transform:

$$\begin{aligned}(F^{-1}(\tilde{F}f))(\boldsymbol{x}) &\approx (F^{-1}(Ff))(\boldsymbol{x}) + i2\pi\varepsilon(F^{-1}(FS))(\boldsymbol{x}) \\ &\approx f(\boldsymbol{x}) + i2\pi\varepsilon f \star (F^{-1}S)(\boldsymbol{x}),\end{aligned}$$

Finally, we obtain that if a signal f is subjected by a small noise we obtain the new signal:

$$\tilde{f}(\boldsymbol{x}) \approx f(\boldsymbol{x}) + i2\pi\varepsilon f \star (F^{-1}S)(\boldsymbol{x}).$$

So the intensity of the signal is modified by the convolution of the signal with the inverse Fourier transform of the Gaussian random field.[1, 5, 11]

This modification implies a not uniform translation of the values associated to the pixels.

An application of the previous remark will be seen better in the following example.

Example 1.3.1. In order to better understand the role of Fourier phase spectrum we consider $f : [0,1]^2 \to \mathbb{R}$ simple function, defined in the following way:

$$f(x,y) = \begin{cases} 0 & \text{if} \quad x \in [0,0.5] \times [0,0.5]; \\ 75 & \text{if} \quad x \in [0,0.5] \times [0.5,1]; \\ 150 & \text{if} \quad x \in [0.5,1] \times [0,0.5]; \\ 225 & \text{if} \quad x \in [0.5,1] \times [0.5,1]; \end{cases}$$

This function assumes as values integer numbers so it describes an image.

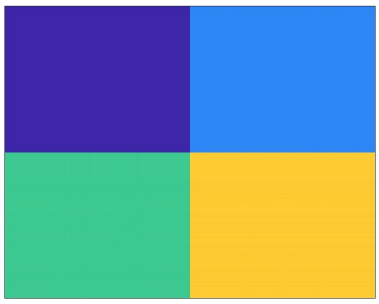


Figure 1.1: Representation of the simple function as an image.

Now we compute it Fourier transform and for each couple (u, v) we multiply $F(u, v)$ by the complex exponential: $e^{i2\pi\langle(u,v),(-0.25,-0.25)\rangle}$. We attain

$$\tilde{F}(u, v) = F(u, v)e^{i2\pi\langle(u,v),(-0.25,-0.25)\rangle}. \tag{1.4}$$

Therefore, we use Theorem 1.2.1 on (1.4) and we get the new function:

$$\tilde{f}(x, y) = \int_{\mathbb{R}^2} F(u, v)e^{i2\pi\langle(u,v),(-0.25,-0,25)\rangle}e^{i2\pi\langle(u,v),(x,y)\rangle}dudv.$$

Finally, we get:

$$\tilde{f}(x, y) = f(x - 0.25, y - 0.25).$$

We can implement this in Matlab, and the result that we obtain is consistent with the theory:

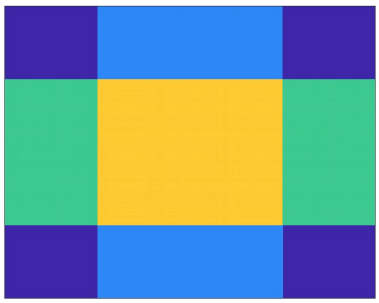


Figure 1.2: Representation of the simple function as an image after the phase modification.

We can also modify the frequency components multiplying them by $e^{i2\pi\epsilon S(u,v)}$, and we know that, if $\epsilon \approx 0$, the result is that

$$\tilde{f}(x,y) \approx f(x,y) + i2\pi\epsilon f \star (F^{-1}S)(x,y).$$

We display also in this case the computation in Matlab and we see the effect of this modification on the signal:

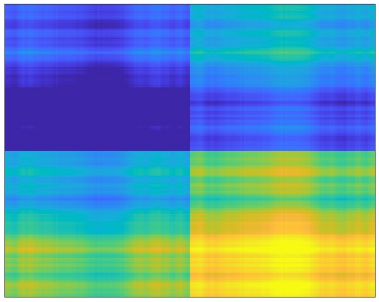


Figure 1.3: Representation of the simple function as an image after a random phase modification.

Now we recall the definition of *Total Variation*.[1]
The Total Variation of an image f is defined as

$$TV(f) = \sum_{|\boldsymbol{x}-\boldsymbol{y}|=l} |f(\boldsymbol{x}) - f(\boldsymbol{y})|, \tag{1.5}$$

where l is the length of a signal. We consider the total variation associated with the functions corresponding to the previous images, pixels have measure 0.005. We find the following result

	Figure 1.1	Figure 1.2	Figure 1.3
TV	9.0000e+04	1.8000e+05	9.4434e+05

.

From this tabular we can observe that a greater modification of the phase of the Fourier components leads to a growth of the Total Variation of the function.

Finally, we can derive the conclusion that the Total Variation is an indicator of how much the image is altered.

This example will justify definitions that we will see in Chapter 4.

Chapter 2

Reproducing Kernel Hilbert Spaces

The main tool we will use for our application is based on reproducing kernel Hilbert space (RKHS). Therefore, let us present the main definitions and results that are essential for a better understanding.

2.1 Scattered data problems

This section is based on the work [4].

In many fields, we can deal with the following problem: we have a set of data (measurements and the respective locations, i.e., where these measurements are taken) and we want to find a rule which allows us to deduce information about the process we are studying also at locations different from those at which we obtained our measurements.

For this kind of problem, our goal is to produce a function s that faithfully represents the given data and allows us to make predictions at other times, locations, or parameter settings. Usually, this function is called *surrogate model*, *simulation metamodel* or a *response surface*.

We should probably be more precise when we say that our model should "faithfully represent the given data". We mean that the function s should match the given measurements at the corresponding locations, or we may be satisfied if the function s simply approximates the given measurements at the corresponding locations. In the former case, we would refer to the process as *(scattered data) interpolation*, while in the latter case it would be *(scattered data) approximation*.

For a more precise mathematical description, we need to set up some notation that we will be using in these chapters. First, we assume that the *measurement locations* (or *data sites*) are labeled $\boldsymbol{x}_j$, $j = 1, \ldots, N$, and the corresponding *measurements* (or *data values*) are the values $f(\boldsymbol{x}_j)$ $j = 1, \ldots, N$, of a certain scalar function f which is the function that de-

scribes exactly the process we are trying to study. We will use $\mathcal{X} \subset \Omega$ to denote the set of data sites and assume that Ω is an open in $\mathbb{R}^d$. We are ready for a precise formulation of the scattered data interpolation problem.

Problem 2.1.1 (**Scattered data problem**). *Given the data* $\{(\boldsymbol{x}_i, f(\boldsymbol{x}_i)\}_{i=1}^N$ *with* $\boldsymbol{x}_i \in \mathbb{R}^d$, $f(\boldsymbol{x}_i) \in \mathbb{R}$, *find a continuous function* s_f *such that* $s_f(\boldsymbol{x}_i) = f(\boldsymbol{x}_i)$, $i = 1, \ldots, N$.

A convenient and common approach to solve the scattered data problem is to assume that the function s_f is a linear combination of certain *basis functions* B_j, i.e.,

$$s_f(\boldsymbol{x}) = \sum_{j=1}^{N} c_j B_j(\boldsymbol{x}) \quad \boldsymbol{x} \in \mathbb{R}^d. \tag{2.1}$$

Solving Problem 2.1.1 means imposing the conditions $s_f(\boldsymbol{x}_i) = f(\boldsymbol{x}_i)$, $i = 1, \ldots, N$, and this leads to a system of linear equations of the form

$$B\boldsymbol{c} = \boldsymbol{f}, \tag{2.2}$$

where the entries of the interpolation matrix are given by $B_{ij} = B_j(\boldsymbol{x}_i)$, $i, j = 1, \ldots, N$, $\boldsymbol{c} = (c_1, \ldots, c_N)^T$, and $\boldsymbol{f} = (f(\boldsymbol{x}_1), \ldots, f(\boldsymbol{x}_N))^T$.

Problem 2.1.1 is well posed if the matrix B is invertible. So our aim is to find basis functions such that B is invertible.

2.1.1 The Haar-Mairhuber-Curtis theorem

Haar systems play a fundamental role in the theory of function approximation by interpolation since the existence of such a system guarantees uniqueness of an interpolant and thus the fact that the problem is well-posed. This is immediately clear from the definition of a Haar system.

Definition 2.1.1 (**Haar system**). Let the finite-dimensional linear function space $\mathcal{B} \subseteq C(\Omega)$ have a basis $B_1, \ldots, B_N$. Then $\mathcal{B}$ is a Haar space on Ω if

$$\det B \neq 0 \tag{2.3}$$

for any set of distinct $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N$ in Ω. Here B is the matrix with entries $B_{ij} = B_j(\boldsymbol{x}_i)$ and the set $\{B_1, \ldots, B_N\}$ is called a *Haar system*.

For $d = 1$ and $\Omega = [a, b]$ the situation is straightforward.

As soon as $d > 1$, one can no longer ensure a unique interpolant. This fact is implied by the *Haar-Mairhuber-Curtis theorem*.

Theorem 2.1.1 (**Haar-Mairhuber-Curtis theorem**). *If* $\Omega \subset \mathbb{R}^d$, $d \geq 2$, *contains an interior point, then there exist no Haar spaces of continuous functions except for trivial ones, i.e., spaces spanned by a single function.*

Note that this theorem contains two notions of "dimensions": the dimension d of the space in which the data sites lie, and the dimension N of the linear function space $\mathcal{B}$.

The Haar-Mairhuber-Curtis theorem implies that one can not fix a nontrivial basis consisting of more than $N = 1$ function and hope that interpolation from that space at N arbitrary (distinct) points $\boldsymbol{x}_i \in \mathbb{R}^d$, $d \geq 2$, has a unique solution.

However, if one selects the basis after the data sites are given, then there are no problems suggested by the theorem. A kind of functions that are usually employed for the composition of bases are called kernel functions.

For us a *kernel* k is a *real-valued symmetric function of two variables*, i.e.,

$$\begin{aligned} k : \Omega \times \Omega &\to \mathbb{R}; \\ (\boldsymbol{x}, \boldsymbol{y}) &\mapsto k(\boldsymbol{x}, \boldsymbol{y}). \end{aligned}$$

Therefore, the matrix K is generated by the kernel k in the following sense.

Initially, we can select the set $\mathcal{B} = \{k(\cdot, \boldsymbol{x}_1), \ldots, k(\cdot, \boldsymbol{x}_N)\}$ as a basis that is naturally adapted to the points $\mathcal{X} = \{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N\}$ so that (2.1) becomes

$$s_f(\boldsymbol{x}) = \sum_{i=1}^{N} c_j k(\boldsymbol{x}, \boldsymbol{x}_j) = \boldsymbol{k}(\boldsymbol{x})^T \boldsymbol{c}, \quad \boldsymbol{x} \in \mathbb{R}^d, \tag{2.4}$$

where $\boldsymbol{k}(\boldsymbol{x})^T = (k(\boldsymbol{x}, \boldsymbol{x}_1), \ldots, k(\boldsymbol{x}, \boldsymbol{x}_N))^T$. Repeating the same reasoning that leads to (2.2) we attain

$$K\boldsymbol{c} = \boldsymbol{f}. \tag{2.5}$$

Assuming K to be invertible, we can find the coefficients as $\boldsymbol{c} = K^{-1}\boldsymbol{f}$ and evaluate the kernel interpolant as

$$s_f(\boldsymbol{x}) = \boldsymbol{k}(\boldsymbol{x})^T K^{-1} \boldsymbol{y}, \tag{2.6}$$

This kernel approach can be placed on a solid theoretical foundation by using *(strictly) positive definite kernels*, so that the matrix K is positive definite, and therefore invertible.

2.1.2 Positive Definite Kernels

Now we consider the equation (2.5), where $K_{ij} = k(\boldsymbol{x}_i, \boldsymbol{x}_j)$, $i, j = 1, \ldots, N$. Linear algebra tells us that the system will have a unique solution whenever K is nonsingular. We will focus primarily on the kernels k that generate *positive definite matrices*, which are known to be nonsingular since all of their eigenvalues are positive.

We begin by presenting the definition of a positive definite matrix K and then relate this idea to the concept of positive definite kernels.

Definition 2.1.2 (**Positive definite matrix**)**.** A real symmetric $N \times N$ matrix K is called *positive definite* if its associated quadratic form is positive for any nonzero coefficient vector $\boldsymbol{c} = (c_1, \ldots, c_N)^T \in \mathbb{R}^N$, i.e.,

$$\boldsymbol{c}^T K \boldsymbol{c} > 0.$$

The matrix is called *positive semi-definite* if the quadratic form is allowed to be nonnegative.

Definition 2.1.3. (Positive definite kernel) A symmetric kernel k is called *positive definite* on $\Omega \times \Omega$ if its associated *kernel matrix* K with entries $K_{ij} = k(\boldsymbol{x}_i, \boldsymbol{x}_j)$, $i, j = 1, \ldots, N$, is positive semi-definite for any $N \in \mathbb{N}$ and for any set $\mathcal{X} = \{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N\} \subset \Omega$ of distinct points.

If we want to disallow the possibility of a zero eigenvalue, we would have to refer to *a strictly positive definite kernel*, in this case the associated kernel matrix K is positive definite.

2.2 Reproducing Kernel Hilbert Space

Definition 2.2.1 (**Reproducing kernel Hilbert space**)**.** Let $\mathcal{H}_k$ be a Hilbert space of real-valued functions f defined on $\Omega \subseteq \mathbb{R}^d$ with inner product $\langle \cdot, \cdot \rangle_{\mathcal{H}_k}$. A kernel $k : \Omega \times \Omega \to \mathbb{R}$ is called *reproducing kernel* for $\mathcal{H}_k$ if

1. $k(\cdot, \boldsymbol{x}) \in \mathcal{H}_k$ for all $x \in \Omega$;
2. $f(x) = \langle f, k(\cdot, \boldsymbol{x}) \rangle_{\mathcal{H}_k}$ for all $f \in \mathcal{H}_k$ and all $\boldsymbol{x} \in \Omega$.

Usually the Hilbert space provided with a reproducing kernel is called *reproducing kernel Hilbert spaces* (RKHS).

The name *reproducing kernel* is motivated by the second property called *reproducing property*, which says that the action of taking the inner product of any function $f \in \mathcal{H}_k$ with $k(\cdot, \boldsymbol{x})$ corresponds to point evaluation of f at $\boldsymbol{x}$.

We prove that the reproducing kernel associated to a Hilbert space is unique.

Theorem 2.2.1. *The reproducing kernel of a RKHS is unique.*

Proof. Let k and k' be two reproducing kernels of the same Hilbert space. Let us consider the two functions $k_{\boldsymbol{x}} = k(\cdot, \boldsymbol{x})$ and $k'_{\boldsymbol{x}} = k'(\cdot, \boldsymbol{x})$, we have

$$\begin{aligned} ||k_{\boldsymbol{x}} - k'_{\boldsymbol{x}}||^2_{\mathcal{H}_k} &= \langle k_{\boldsymbol{x}} - k'_{\boldsymbol{x}}, k_{\boldsymbol{x}} - k'_{\boldsymbol{x}} \rangle_{\mathcal{H}_k} \\ &= \langle k_{\boldsymbol{x}} - k'_{\boldsymbol{x}}, k_{\boldsymbol{x}} \rangle_{\mathcal{H}_k} - \langle k_{\boldsymbol{x}} - k'_{\boldsymbol{x}}, k'_{\boldsymbol{x}} \rangle_{\mathcal{H}_k} \\ &= k_{\boldsymbol{x}}(\boldsymbol{x}) - k'_{\boldsymbol{x}}(\boldsymbol{x}) - k_{\boldsymbol{x}}(\boldsymbol{x}) + k'_{\boldsymbol{x}}(\boldsymbol{x}) = 0 \end{aligned}$$

then $k_{\boldsymbol{x}} = k'_{\boldsymbol{x}}$ and finally:

$$k_{\boldsymbol{x}}(\boldsymbol{x}, \boldsymbol{y}) = k'_{\boldsymbol{x}}(\boldsymbol{x}, \boldsymbol{y}).$$

□

Now we give some properties of RKHS.

Proposition 2.2.1 (Properties of RKHS). *Let H_k be a RKHS on $\Omega \subset \mathbb{R}^d$ then*

1. $k(\boldsymbol{x}, \boldsymbol{y}) = \langle k(\cdot, \boldsymbol{y}), k(\cdot, \boldsymbol{x}) \rangle_{\mathcal{H}_k}$;
2. *Hilbert space norm convergence implies pointwise convergence:*

$$||f - f_n||_{\mathcal{H}_k} \to 0 \Longrightarrow |f(\boldsymbol{x}) - f_n(\boldsymbol{x})| \to 0 \quad \forall \boldsymbol{x} \in \Omega;$$

$$(|f(\boldsymbol{x}) - f_n(\boldsymbol{x})| = \langle f - f_n, k(\cdot, \boldsymbol{x}) \rangle_{\mathcal{H}_k} \leq ||f - f_h||_{\mathcal{H}_k} ||k(\cdot, \boldsymbol{x})||_{\mathcal{H}_k})$$

3. *Each linear subspace of a RKHS it is itself a RKHS.*

Someone could wonder about the existence of a reproducing kernel associated to a certain Hilbert space. The following theorem can answer this question.

Theorem 2.2.2. *A Hilbert space $\mathcal{H}$ of functions on Ω has a reproducing kernel if and only if the evaluation functional at any point in Ω is continuous on $\mathcal{H}$.*

Furthermore, we observe that we can easily compute the norm of certain functions that belong to the RKHS.

Observation 2.2.1. The definition of a reproducing kernel Hilbert space tells us that $\mathcal{H}_k(\Omega)$ contains all functions of the form

$$f = \sum_{i=1}^{N} c_j k(\cdot, \boldsymbol{x}_j)$$

provided $\boldsymbol{x}_j \in \Omega$ (and N is arbitrary). Using the properties of reproducing kernel Hilbert spaces along with this special form of f we have that

$$\begin{aligned} ||f||^2_{\mathcal{H}_k} = \langle f, f \rangle_{H_k} &= \left\langle \sum_{i=1}^{N} c_i K(\cdot, \boldsymbol{x}_i), \sum_{j=1}^{N} c_j k(\cdot, \boldsymbol{x}_j) \right\rangle_{H_k} \\ &= \sum_{i=1}^{N} \sum_{j=1}^{N} c_i c_j \left\langle k(\cdot, \boldsymbol{x}_i), k(\cdot, \boldsymbol{x}_j) \right\rangle_{H_k} \\ &= \sum_{i=1}^{N} \sum_{j=1}^{N} c_i c_j k(\boldsymbol{x}_i, \boldsymbol{x}_j) = \boldsymbol{c}^T K \boldsymbol{c} \end{aligned}$$

Thus, for these very special types of f, we can easily calculate the Hilbert space norm, and this result will be very important in Chapter 3.

In particular, if $f = s_f$, we have $\boldsymbol{c} = K^{-1}\boldsymbol{f}$ and then

$$||s_f||^2_{\mathcal{H}_k} = \boldsymbol{f}^T K^{-T} K K^{-1} \boldsymbol{f} = \boldsymbol{f}^T K^{-1} \boldsymbol{f}$$

From what we see, the norm of the kernel interpolant depends on the given data.

We conclude this section talking about *native spaces* and presenting two examples.

Let k be an spd kernel on $\Omega \times \Omega$. We define the vector space

$$\tilde{\mathcal{N}_k} = \text{span}\{k(\cdot, \boldsymbol{x}), \boldsymbol{x} \in \Omega\},$$

provided with the bilinear form defined in the following way:

$$\left\langle \sum_{i=1}^{N_k} c_i k(\cdot, \boldsymbol{x}_i), \sum_{j=1}^{N_k} c_j k(\cdot, \boldsymbol{x}_j) \right\rangle_{\tilde{\mathcal{N}_k}} := \sum_i \sum_j c_i c_j k(\boldsymbol{x}_i, \boldsymbol{x}_j). \tag{2.7}$$

We can observe that the vector space has cardinality equal to Ω so $\tilde{\mathcal{N}_k}$ could contain an infinite number of elements. Then

Theorem 2.2.3. *Let k be a s.p.d. kernel on $\Omega \times \Omega$.*

Then, $\langle \cdot, \cdot \rangle_{\tilde{\mathcal{N}_k}}$ defines an inner product, and so $\tilde{\mathcal{N}_k}$ is a pre-Hilbert space with reproducing kernel k.

The completion of this vector space with respect to the norm induced by the inner product has a name:

Definition 2.2.2. We call *native space* $\mathcal{N}_k$, the completion of $\tilde{\mathcal{N}_k}$, with respect to the norm $|| \cdot ||_{\tilde{\mathcal{N}_k}}$, such that:

$$||f||_{\tilde{\mathcal{N}_k}} = ||f||_{\mathcal{N}_k}.$$

From the previous theorem we have that k is the reproducing kernel of $\mathcal{N}_k$, so $\mathcal{N}_k$ is RKHS.

In order to fix the ideas we present two strictly positive definite kernels that will be used also in the application.

Example 2.2.1. The first kernel function that we are going to introduce is probably one of the most famous radial functions: the Gaussian kernel. It is defined as

$$k_G(\boldsymbol{x}, \boldsymbol{y}) = e^{-\varepsilon ||\boldsymbol{x}-\boldsymbol{y}||^2}, \tag{2.8}$$

and it can be also expressed in its radial form:

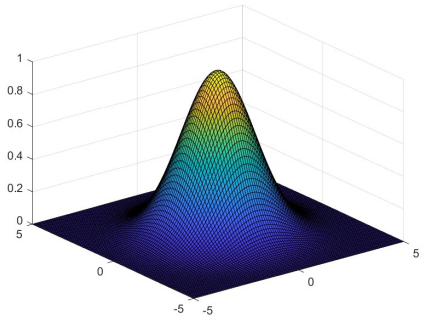
$$k_G(r) = e^{-\varepsilon r^2}, \tag{2.9}$$

where ε is called *shape parameter* because it determines the velocity of the radial decrease.

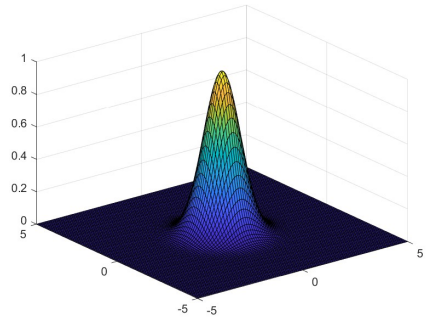
This can be seen from its radial derivative:

$$\frac{d}{dr}k_G(r) = -\varepsilon 2re^{-\epsilon r^2} < 0,$$

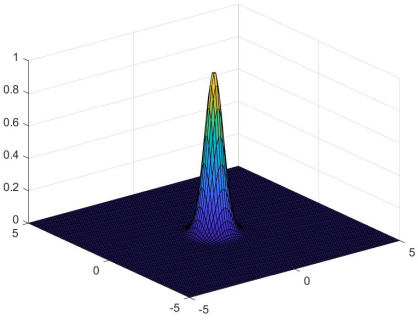
and we can clearly notice that the absolute value of this derivative increase if ε increase. In Figure 2.1 we show the geometrical representation of the dependence of Gaussian kernel on the shape parameter ε:



(a) Gaussian kernel with $\varepsilon = 0.5$.



(b) Gaussian kernel with $\varepsilon = 1$.



(c) Gaussian kernel with $\varepsilon = 2$.

Figure 2.1: Comparison of the Gaussian kernel with different shape parameters.

We are interested in showing that it is a strictly positive definite kernel. In order to obtain this we have to prove that it is symmetric and that the

associated kernel matrix on the data $\mathcal{X} = \{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N\}$, i.e., $K = k_G(\boldsymbol{x}_i, \boldsymbol{x}_j)$ is positive definite, so we have to ensure $\boldsymbol{a}^T K \boldsymbol{a} > 0$ for any non zero $\boldsymbol{a}$.

We show these two properties for the Gaussian kernel:

1. The Gaussian kernel is symmetric because the distance $d(\boldsymbol{x}, \boldsymbol{y}) = ||\boldsymbol{x} - \boldsymbol{y}||$ is symmetric:

$$k_G(\boldsymbol{x}, \boldsymbol{y}) = e^{-\varepsilon||\boldsymbol{x}-\boldsymbol{y}||^2} = e^{-\varepsilon||\boldsymbol{y}-\boldsymbol{x}||^2} = k_G(\boldsymbol{y}, \boldsymbol{x}).$$

2. The Gaussian kernel is s.p.d. because

$$a^T K a = \sum_{i=1}^{N}\sum_{j=1}^{N} a_i a_j K(\boldsymbol{x}_i, \boldsymbol{x}_j) = \sum_{i=1}^{N}\sum_{j=1}^{N} a_i a_j e^{-\varepsilon||\boldsymbol{x}_i-\boldsymbol{x}_j||^2} > 0 \quad \forall a \in \mathbb{R}^N.$$

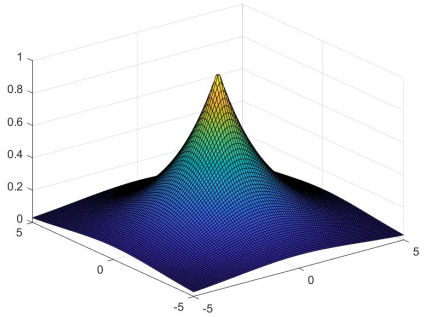
   Now we can apply the Theorem 2.2.3 and obtain a RKHS with reproducing kernel the Gaussian kernel k_G.

Example 2.2.2. The second kernel we are talking about is the Matérn C_0 kernel. It is defined as

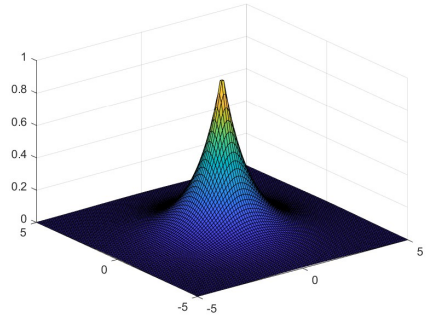
$$k_M(\boldsymbol{x}, \boldsymbol{y}) = e^{-\varepsilon||\boldsymbol{x}-\boldsymbol{y}||}, \tag{2.10}$$

and we can repeat the same steps as above in order to see that it is a strictly positive definite kernel with an associated RKHS.

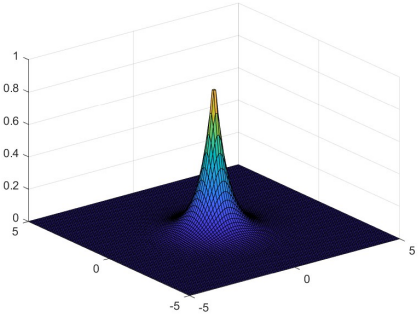
In Figure 2.2 we give also for the Matérn kernel the geometric visualization as the shape parameter changes.



(a) Matérn kernel with $\varepsilon = 0.5$.



(b) Matérn kernel with $\varepsilon = 1$.



(c) Matérn kernel with $\varepsilon = 2$.

Figure 2.2: Comparison of the Matérn kernel with different shape parameters.

2.3 Fill Distance and Error Estimates

Given a set of data $\mathcal{X} = \{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N\}$, and a research space Ω, a tool that is usually adopted in approximation theory to establish a bound of the error between the unknown function and the interpolant is called *fill distance*

$$h \equiv h_{\mathcal{X},\Omega} := \sup_{\boldsymbol{x}\in\Omega} \min_{\boldsymbol{x}_j\in\mathcal{X}} ||\boldsymbol{x} - \boldsymbol{x}_j||.$$

The fill distance indicates the highest distance between the smallest distances between the points of $\mathcal{X}$ and the points of $\mathcal{X}$.

Therefore, we are interested in whether the error

$$||f - \mathcal{P}_f^{(h)}||_\infty$$

tends to zero as $h \to 0$, and if so, how fast. Here $\{\mathcal{P}_f^{(h)}\}_h$ presents a sequence of interpolation (or, more generally, projection) operators that vary with the fill distance h.

Since we want to employ the properties of reproducing kernel Hilbert spaces presented in this chapter we will concentrate on error estimates for functions $f \in \mathcal{N}_k$.

The term that is often used to measure the speed of convergence to zero is approximation order. We say that the approximation operator $\mathcal{P}_f^{(h)}$ has *L_p-approximation order k* if

$$||f - \mathcal{P}_f^{(h)}||_p = \mathcal{O}(j^k) \quad \text{for} \quad h \to 0.$$

Moreover, if we can also show that $||f - \mathcal{P}_f^{(h)||_p} \neq o(j^k)$, then $\mathcal{P}^{(h)}$ has *exact L_p-approximation order k*. We will concentrate mostly in the case $p = \infty$.

2.3.1 Lagrange Form of the Interpolant and Cardinal Basis Functions

The key idea for the following discussion is to express the interpolant in *Lagrange form*, i.e., as a linear combination of polynomials which assume the value 1 on a specific interpolation node and 0 on the other nodes , using the so-called *cardinal basis functions*.

In the previous section, we have defined strictly positive definite kernel k and seen that the linear system

$$K\boldsymbol{c} = \boldsymbol{y}$$

with $K_{ij} = k(\boldsymbol{x}_i, \boldsymbol{x}_j)$, $i, j = 1, \ldots, N$, $\boldsymbol{c} = [c_1, \ldots, c_N]^T$, and
$\boldsymbol{y} = [f(\boldsymbol{x}_1), \ldots, f(\boldsymbol{x}_N)]^T$ has a unique solution.

In order to obtain the cardinal basis function u_j^*, $j = 1, \ldots, N$, with the Lagrange property $u_j^*(\boldsymbol{x}_i) = \delta_{ij}$, we consider the linear system

$$K\boldsymbol{u}^*(\boldsymbol{x}) = k(\boldsymbol{x}), \tag{2.11}$$

where K is as above, $\boldsymbol{u}^* = [u_1^*, \ldots, u_N^*]^T$ and $\boldsymbol{k} = [k(\cdot, \boldsymbol{x}_1), \ldots, k(\cdot, \boldsymbol{x}_N)]^T$. Thus,

Theorem 2.3.1. *Suppose k is a strictly positive definite kernel on $\mathbb{R}^s$. Then, for any distinct points $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N$, there exist functions $u_j^* \in span\{k(\cdot, \boldsymbol{x}_j), j = 1, \ldots, N\}$ such that $u_j^* = \delta_{ij}$.*

Therefore, we can write the interpolant $\mathcal{P}_f$ to f at $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N$ in the cardinal form

$$\mathcal{P}_f(\boldsymbol{x}) = \sum_{i=1}^{N} f(\boldsymbol{x}_j) u_j^*(\boldsymbol{x}) \quad \boldsymbol{x} \in \mathbb{R}^s. \tag{2.12}$$

It is of interest to note that the cardinal functions do not depend on the data values of the interpolation problem. Once the data sites are fixed and the basic function is chosen with an appropriate shape parameter, the cardinal functions are determined by the linear system (2.11).

2.3.2 The Power Function

We need another important ingredient for our error estimates: the so called *power function*. To this end, we consider a domain $\Omega \subseteq \mathbb{R}^s$. Then for any strictly positive definite kernel $k \in \mathcal{C}(\Omega \times \Omega)$, any set of distinct points $\mathcal{X} = \{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N\} \subseteq \Omega$, and any vector $\boldsymbol{u} \in \mathbb{R}^N$, we define the quadratic form

$$Q(\boldsymbol{u}) = k(\boldsymbol{x}, \boldsymbol{x}) - 2\sum_{j=1}^{N} u_j k(\boldsymbol{x}, \boldsymbol{x}_j) + \sum_{j=1}^{N}\sum_{j=1}^{N} u_i u_j k(\boldsymbol{x}_i, \boldsymbol{x}_j). \tag{2.13}$$

Then

Definition 2.3.1. Suppose $\Omega \subseteq \mathbb{R}^s$ and $k \in \mathcal{C}(\Omega \times \Omega)$ is strictly positive definite on $\mathbb{R}^s$. For any distinct points $\mathcal{X} = \{\boldsymbol{x}_1, \ldots \boldsymbol{x}_N\} \subseteq \Omega$ the *power function* is defined by

$$[P_{k,\mathcal{X}}(\boldsymbol{x})]^2 = Q(\boldsymbol{u}^*(\boldsymbol{x})) \tag{2.14}$$

where $\boldsymbol{u}^*$ is the vector of cardinal functions from Theorem 2.3.1.

The following remark will be useful for the proof of Theorem 2.3.2.

Observation 2.3.1. Using the definition of the inner product (2.7) and of

reproducing kernel we can rewrite the quadratic form $Q(\boldsymbol{u})$ as

$$\begin{aligned}
Q(\boldsymbol{u}) &= k(\boldsymbol{x},\boldsymbol{x}) - 2\sum_{j=1}^{N} u_j k(\boldsymbol{x},\boldsymbol{x}_j) + \sum_{i=1}^{N}\sum_{j=1}^{N} u_i u_j k(\boldsymbol{x}_i,\boldsymbol{x}_j) \\
&= \langle k(\cdot,\boldsymbol{x}), k(\cdot,\boldsymbol{x})\rangle_{\mathcal{N}_k} - 2\sum_{j=1}^{N} u_j \langle k(\cdot,\boldsymbol{x}), k(\cdot,\boldsymbol{x}_j)\rangle_{\mathcal{N}_k} \\
&+ \sum_{i=1}^{N}\sum_{j=1}^{N} u_i u_j \langle k(\cdot,\boldsymbol{x}_i), k(\cdot,\boldsymbol{x}_j)\rangle_{\mathcal{N}_k} \\
&= \langle k(\cdot,\boldsymbol{x}), k(\cdot,\boldsymbol{x})\rangle_{\mathcal{N}_k} - \langle k(\cdot,\boldsymbol{x}), \sum_{j=1}^{N} u_j k(\cdot,\boldsymbol{x}_j)\rangle \\
&- \langle k(\cdot,\boldsymbol{x}), \sum_{j=1}^{N} u_j k(\cdot,\boldsymbol{x}_j)\rangle + \langle \sum_{j=1}^{N} u_j k(\cdot,\boldsymbol{x}_j), \sum_{j=1}^{N} u_j k(\cdot,\boldsymbol{x}_j)\rangle \\
&= \langle k(\cdot,\boldsymbol{x}) - \sum_{j=1}^{N} u_j k(\cdot,\boldsymbol{x}_j), k(\cdot,\boldsymbol{x}) - \sum_{j=1}^{N} u_j k(\cdot,\boldsymbol{x}_j)\rangle_{\mathcal{N}_k} \\
&= \left\| k(\cdot,\boldsymbol{x}) - \sum_{j=1}^{N} u_j k(\cdot,\boldsymbol{x}_j) \right\|_{\mathcal{N}_k}^2 .
\end{aligned}$$

Furthermore, using the linear system notation employed earlier, i.e., $K_{ij} = k(\boldsymbol{x}_i,\boldsymbol{x}_j)$, $i,j = 1,\ldots,N$, $\boldsymbol{u} = [u_1,\ldots,u_N]^T$, and $\boldsymbol{k} = [k(\cdot,\boldsymbol{x}_1),\ldots,k(\cdot,\boldsymbol{x}_N)]^T$, we have alternative representations of the power function. Using the matrix-vector notation for $Q(\boldsymbol{u})$, the power function is given as

$$P_{k,\mathcal{X}}(\boldsymbol{x}) = \sqrt{Q(\boldsymbol{u}^*(\boldsymbol{x}))} = \sqrt{k(\boldsymbol{x},\boldsymbol{x}) - 2(\boldsymbol{u}^*(\boldsymbol{x}))^T\boldsymbol{k}(\boldsymbol{x}) + (\boldsymbol{u}^*)^T K\boldsymbol{u}^*(\boldsymbol{x})}.$$

However, by the definition of the cardinal function $K\boldsymbol{u}^*(\boldsymbol{x}) = \boldsymbol{k}(\boldsymbol{x})$, we have the two new variants

$$P_{k,\mathcal{X}}(\boldsymbol{x}) = \sqrt{k(\boldsymbol{x},\boldsymbol{x}) - (\boldsymbol{u}^*)^T\boldsymbol{k}(\boldsymbol{x})} = \sqrt{k(\boldsymbol{x},\boldsymbol{x}) - (\boldsymbol{u}^*(\boldsymbol{x}))^T K\boldsymbol{u}^*(\boldsymbol{x})}.$$

These formulas can be used for the numerical evaluation of the power function at $\boldsymbol{x}$. To this end one has to first find the value of the cardinal functions $\boldsymbol{u}^*(\boldsymbol{x})$ by solving the system $K\boldsymbol{u}^*(\boldsymbol{x}) = \boldsymbol{k}(\boldsymbol{x})$. This results in

$$P_{k,\mathcal{X}}(\boldsymbol{x}) = \sqrt{k(\boldsymbol{x},\boldsymbol{x}) - (\boldsymbol{k}(\boldsymbol{x}))^T K^{-1}\boldsymbol{k}(\boldsymbol{x})}. \tag{2.15}$$

Since K is a positive definite matrix, in the previous equality we are subtracting a strictly positive quantity, so we obtain:

$$0 \leq P_{k,\mathcal{X}}(\boldsymbol{x}) \leq \sqrt{k(\boldsymbol{x},\boldsymbol{x})}$$

2.3.3 Error Estimates for Functions in $\mathcal{N}_k$

Now we can give a first generic error estimate.

Theorem 2.3.2. *Let $\Omega \subseteq \mathbb{R}^s$, $k \in \mathcal{C}(\Omega, \Omega)$ be strictly positive definite on $\mathbb{R}^s$, and suppose that the points $\mathcal{X} = \{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N\}$ are distinct. Denote the interpolant to $f \in \mathcal{N}_k$ on $\mathcal{X}$ by $\mathcal{P}_f$. Then for every $\boldsymbol{x} \in \Omega$*

$$|f(\boldsymbol{x}) - \mathcal{P}_f(\boldsymbol{x})| \leq P_{k,\mathcal{X}}(\boldsymbol{x})||f||_{\mathcal{N}_k}. \tag{2.16}$$

Proof. Since f is assumed to lie in the native space of k, the reproducing property of k yields

$$f(\boldsymbol{x}) = \langle f, k(\cdot, \boldsymbol{x})\rangle_{\mathcal{N}_k}.$$

We express the interpolant in its cardinal form and apply the reproducing property of k. This gives us

$$\begin{aligned}\mathcal{P}_f(\boldsymbol{x}) &= \sum_{j=1}^{N} f(x_j)u_j^*(\boldsymbol{x}) \\ &= \sum_{j=1}^{N} u_j^*(\boldsymbol{x})\langle f, k(\cdot, \boldsymbol{x}_j)\rangle_{\mathcal{N}_k} \\ &= \langle f, \sum_{j=1}^{N} u_j^*(\boldsymbol{x})k(\cdot, \boldsymbol{x}_j)\rangle_{\mathcal{N}_k}.\end{aligned}$$

Now all that remains to be done is to combine the two formulas just derived and apply the Cauchy-Schwartz inequality. Thus,

$$\begin{aligned}|f(\boldsymbol{x}) - \mathcal{P}_f(\boldsymbol{x})| &= \left|\langle f, k(\cdot, \boldsymbol{x}) - \sum_{j=1}^{N} u_j^*(\boldsymbol{x})k(\cdot, \boldsymbol{x}_j)\rangle_{\mathcal{N}_k}\right| \\ &\leq ||f||_{\mathcal{N}_k} P_{k,\mathcal{X}}(\boldsymbol{x}),\end{aligned}$$

where we have used Observation 2.3.1 and the definition of power function. □

2.3.4 Error Estimates in Terms of the Fill Distance

The next step is to refine this error bound of Theorem 2.3.2 by expressing the influence of the data locations in terms of the fill distance. The most common strategy of obtaining error bounds in numerical analysis is to take advantage of the polynomial precision of a method (at least locally), and then to apply a Taylor expansion. With this in mind we observe

Theorem 2.3.3. *Let $\Omega \subseteq \mathbb{R}^s$, and suppose $k \in \mathcal{C}(\Omega \times \Omega)$. Let $\mathcal{X} = \{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N\}$, be a set of distinct points in Ω, and define the quadratic form $Q(\boldsymbol{u})$ as in (2.13). The minimum of $Q(\boldsymbol{u})$ is given for the vector $\boldsymbol{u} = \boldsymbol{u}^*(\boldsymbol{x})$ from Theorem 2.3.1, i.e.,*

$$Q(\boldsymbol{u}^*(\boldsymbol{x})) \leq Q(\boldsymbol{u}) \quad \forall \boldsymbol{u} \in \mathbb{R}^N$$

Proof. We showed above that

$$Q(\boldsymbol{u}) = k(\boldsymbol{x}, \boldsymbol{x}) - 2\boldsymbol{u}^T \boldsymbol{k}(\boldsymbol{x}) + \boldsymbol{u}^T K \boldsymbol{u}.$$

The minimum of this quadratic form is given by the solution of the linear system

$$K\boldsymbol{u} = \boldsymbol{k}(\boldsymbol{x}).$$

This, however, is given by the cardinal functions $\boldsymbol{u} = \boldsymbol{u}^*(\boldsymbol{x})$.

□

In the proof below we will use a special coefficient vector $\tilde{\boldsymbol{u}}$ which provides the polynomial precision desired for the proof of the refined error estimate. Theorem 2.3.4 requires the notion of a domain that satisfies an interior cone condition.

Definition 2.3.2. A region $\Omega \subseteq \mathbb{R}^s$ satisfies an *interior cone condition* if there exists an angle $\theta \in (0, \frac{\pi}{2})$ and a radius $r > 0$ such that for every $\boldsymbol{x} \in \Omega$ there exists a unit vector $\boldsymbol{\xi}(\boldsymbol{x})$ such that the cone

$$C = \{\boldsymbol{x} + \lambda \boldsymbol{y} : \boldsymbol{y} \in \mathbb{R}^s, ||\boldsymbol{y}||_2 = 1, \boldsymbol{y}^T \boldsymbol{\xi}(\boldsymbol{x}) \geq \cos\theta, \lambda \in [0, r]\}$$

is contained in Ω.

Existence of an approximation scheme with local polynomial precision is guaranteed by

Theorem 2.3.4. *Suppose $\Omega \subseteq \mathbb{R}^s$ is bounded and satisfies an interior cone condition, and let l be a non-negative integer. Then there exist positive constants h_0, c_1 and c_2 such that for all $\mathcal{X} = \{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N\} \subseteq \Omega$ with $h_{\mathcal{X},\Omega} \leq h_0$ and every $\boldsymbol{x} \in \Omega$ there exist numbers $\tilde{u}_1(\boldsymbol{x}), \ldots, \tilde{u}_N(\boldsymbol{x})$ with*

1. *$\sum_{j=1}^N \tilde{u}_j(\boldsymbol{x}) p(\boldsymbol{x}_j) = p(\boldsymbol{x})$, for all polynomials $p \in \Pi_l^s$, i.e., polynomials of power l on dimension s;*
2. *$\sum_{j=1}^N |\tilde{u}_j(\boldsymbol{x})| \leq c_1$,*
3. *$\tilde{u_j}(\boldsymbol{x}) = 0$ if $||\boldsymbol{x} - \boldsymbol{x}_j||_2 \geq c_2 h_{\mathcal{X},\Omega}$.*

A consequence of the interior cone condition is the fact that a domain that satisfies this condition contains balls of a controllable radius.

In the following theorem we will use multi-index notation and multivariate Taylor expansions. For $\boldsymbol{\beta} = (\beta_1, \ldots, \beta_s) \in \mathbb{N}_0^s$ with $|\boldsymbol{\beta}| = \sum_{i=1}^s \beta_i$ we define the differential operators $D^{\boldsymbol{\beta}}$ as

$$D^{\boldsymbol{\beta}} = \frac{\partial^{|\boldsymbol{\beta}|}}{(\partial x_1)^{\beta_1} \ldots (\partial x_s)^{\beta_s}},$$

and the notation $D_2^{\boldsymbol{\beta}} k(\boldsymbol{w}, \cdot)$ indicates that the operator is applied to $k(\boldsymbol{w}, \cdot)$ viewed as a function of the second variable.

The multivariate Taylor expansion of the function $k(\boldsymbol{w}, \cdot)$ centered at $\boldsymbol{w}$ is given by

$$k(\boldsymbol{w}, \boldsymbol{z}) = \sum_{|\boldsymbol{\beta}|<2k} \frac{D_2^{\boldsymbol{\beta}} k(\boldsymbol{w}, \boldsymbol{w})}{\boldsymbol{\beta}!} (\boldsymbol{z} - \boldsymbol{w})^{\boldsymbol{\beta}} + R(\boldsymbol{w}, \boldsymbol{z}),$$

with remainder

$$R(\boldsymbol{w}, \boldsymbol{z}) = \sum_{|\boldsymbol{\beta}|=2k} \frac{D_2^{\boldsymbol{\beta}} k(\boldsymbol{w}, \xi_{\boldsymbol{w},\boldsymbol{z}})}{\boldsymbol{\beta}!} (\boldsymbol{z} - \boldsymbol{w})^{\boldsymbol{\beta}},$$

where $\xi_{\boldsymbol{w},\boldsymbol{z}}$ lies somewhere on the line segment connecting $\boldsymbol{w}$ and $\boldsymbol{z}$. The generic error estimate of Theorem 2.3.2 can now be formulated in terms of the fill distance.

Theorem 2.3.5. *Suppose $\Omega \subseteq \mathbb{R}^s$ is bounded and satisfies an interior cone condition. Suppose $k \in \mathcal{C}^{2k}(\Omega \times \Omega)$ is symmetric and strictly positive definite. Denote the interpolant to $f \in \mathcal{N}_k$ on the set $\mathcal{X}$ by $\mathcal{P}_f$. Then there exist positive constants h_0 and C (independent of $\boldsymbol{x}$, f and k) such that*

$$|f(\boldsymbol{x}) - \mathcal{P}_f(\boldsymbol{x})| \leq C h_{\mathcal{X},\Omega}^k \sqrt{C_k(\boldsymbol{x})} ||f||_{\mathcal{N}_k(\Omega)},$$

provided $h_{\mathcal{X},\Omega} \leq h_0$. Here

$$C_k(\boldsymbol{x}) = \max_{|\boldsymbol{\beta}|=2k} \max_{\boldsymbol{w},\boldsymbol{z} \in \Omega \cap B(\boldsymbol{x}, c_2 h_{\mathcal{X},\Omega})} |D_2^{\boldsymbol{\beta}} k(\boldsymbol{w}, \boldsymbol{z})|,$$

with $B(\boldsymbol{x}, c_2 h_{\mathcal{X},\Omega})$ denoting the ball of radius $c_2 h_{\mathcal{X},\Omega}$ centered at $\boldsymbol{x}$.

Proof. By Theorem 2.3.2 we know

$$|f(\boldsymbol{x}) - \mathcal{P}_f(\boldsymbol{x})| \leq P_{k,\mathcal{X}}(\boldsymbol{x}) ||f||_{\mathcal{N}_k(\Omega)}.$$

Therefore, we now derive the bound

$$P_{k,\mathcal{X}}(\boldsymbol{x}) \leq C h_{\mathcal{X},\Omega}^k \sqrt{C_k(\boldsymbol{x})}$$

for the power function in terms of the fill distance. We know that the power function is defined by

$$[P_{k,\mathcal{X}}(\boldsymbol{x})]^2 = Q(\boldsymbol{u}^*(\boldsymbol{x})).$$

Moreover, we know from Theorem 2.3.3 that the quadratic form $Q(\boldsymbol{u})$ is minimized by $\boldsymbol{u} = \boldsymbol{u}^*(\boldsymbol{x})$. Therefore, any other coefficient vector u will yield an upper bound on the power function.

We take $\boldsymbol{u} = \tilde{\boldsymbol{u}}(\boldsymbol{x})$ from Theorem 2.3.4 so that we are ensured to have polynomial precision of degree $l \geq 2k-1$.

For this specific choice of coefficients we have

$$[P_{k,\mathcal{X}}(\boldsymbol{x})]^2 \leq Q(\boldsymbol{u}) = k(\boldsymbol{x},\boldsymbol{x}) - 2\sum_j u_j k(\boldsymbol{x},\boldsymbol{x}_j) + \sum_i \sum_j u_i u_j k(\boldsymbol{x}_i,\boldsymbol{x}_j),$$

where the sums are over those indices j with $u_j \neq 0$. Now we apply the Taylor expansion to $Q(u)$ centered at $\boldsymbol{x}$ to $k(\boldsymbol{x},\cdot)$ and centered at $\boldsymbol{x}_i$ to $k(\boldsymbol{x}_i,\cdot)$, and evaluate both functions at $\boldsymbol{x}_j$. This yields

$$\begin{aligned} Q(\boldsymbol{u}) = k(\boldsymbol{x},\boldsymbol{x}) - 2\sum_j u_j \left[\sum_{|\boldsymbol{\beta}|<2k} \frac{D_2^{\boldsymbol{\beta}} k(\boldsymbol{x},\boldsymbol{x})}{\boldsymbol{\beta}!}(\boldsymbol{x}_j - \boldsymbol{x})^{\boldsymbol{\beta}} + R(\boldsymbol{x},\boldsymbol{x}_j) \right] \\ + \sum_i \sum_j u_i u_j \left[\sum_{|\boldsymbol{\beta}|<2k} \frac{D_2^{\boldsymbol{\beta}} k(\boldsymbol{x}_i,\boldsymbol{x}_i)}{\boldsymbol{\beta}!}(\boldsymbol{x}_j - \boldsymbol{x}_i)^{\boldsymbol{\beta}} + R(\boldsymbol{x}_i,\boldsymbol{x}_j) \right] \end{aligned} \tag{2.17}$$

Now, we use Theorem 2.3.4 in order to simplify (2.17), we get

$$\sum_j u_j \left[\sum_{|\boldsymbol{\beta}|<2k} \frac{D_2^{\boldsymbol{\beta}} k(\boldsymbol{x},\boldsymbol{x})}{\boldsymbol{\beta}!}(\boldsymbol{x}_j - \boldsymbol{x})^{\boldsymbol{\beta}} \right] = k(\boldsymbol{x},\boldsymbol{x}), \tag{2.18}$$

because we identify $p(\boldsymbol{z}) = \sum_{|\boldsymbol{\beta}|<2k} \frac{D_2^{\boldsymbol{\beta}} k(\boldsymbol{x},\boldsymbol{x})}{\boldsymbol{\beta}!}(\boldsymbol{z} - \boldsymbol{x})^{\boldsymbol{\beta}}$ which is equal to $k(\boldsymbol{x},\boldsymbol{x})$ when $\boldsymbol{\beta} = 0$, because all the other member of the sum are zero if $\boldsymbol{z} = \boldsymbol{x}$ and $\boldsymbol{\beta} > 1$, and

$$\sum_j u_j \left[\sum_{|\boldsymbol{\beta}|<2k} \frac{D_2^{\boldsymbol{\beta}} k(\boldsymbol{x}_i,\boldsymbol{x}_i)}{\boldsymbol{\beta}!}(\boldsymbol{x}_j - \boldsymbol{x}_i)^{\boldsymbol{\beta}} \right] = \sum_{|\boldsymbol{\beta}|<2k} \frac{D_2^{\boldsymbol{\beta}} k(\boldsymbol{x}_i,\boldsymbol{x}_i)}{\boldsymbol{\beta}!}(\boldsymbol{x} - \boldsymbol{x}_i)^{\boldsymbol{\beta}}. \tag{2.19}$$

Now we substitute (2.18) and (2.19) in (2.17) and we obtain:

$$\begin{aligned} Q(\boldsymbol{u}) = k(\boldsymbol{x},\boldsymbol{x}) - 2k(\boldsymbol{x},\boldsymbol{x}) - 2\sum_j u_j R(\boldsymbol{x},\boldsymbol{x}_j) \\ + \sum_i u_i \sum_{|\boldsymbol{\beta}|<2k} \frac{D_2^{\boldsymbol{\beta}} k(\boldsymbol{x}_i,\boldsymbol{x}_i)}{\boldsymbol{\beta}!}(\boldsymbol{x} - \boldsymbol{x}_i)^{\boldsymbol{\beta}} + \sum_i \sum_j u_i u_j R(\boldsymbol{x}_i,\boldsymbol{x}_j). \end{aligned} \tag{2.20}$$

Now we can apply Taylor expansion again and make the observation that

$$\sum_{|\boldsymbol{\beta}|<2k} \frac{D_2^{\boldsymbol{\beta}} k(\boldsymbol{x}_i, \boldsymbol{x}_i)}{\boldsymbol{\beta}!}(\boldsymbol{x} - \boldsymbol{x}_i)^{\boldsymbol{\beta}} = k(\boldsymbol{x}_i, \boldsymbol{x}) - R(\boldsymbol{x}_i, \boldsymbol{x}). \tag{2.21}$$

if we use (2.21) and rearrange the terms in (2.20) we get

$$Q(\boldsymbol{u}) = -k(\boldsymbol{x}, \boldsymbol{x}) - \sum_j u_j \left[2R(\boldsymbol{x}, \boldsymbol{x}_j) - \sum_i u_i R(\boldsymbol{x}_i, \boldsymbol{x}_j) \right] + \sum_i u_i [k(\boldsymbol{x}_i, \boldsymbol{x}) - R(\boldsymbol{x}_i, \boldsymbol{x})] \tag{2.22}$$

One final Taylor expansion we need is (using the symmetry of k)

$$k(\boldsymbol{x}_i, \boldsymbol{x}) = k(\boldsymbol{x}, \boldsymbol{x}_i) = \sum_{|\boldsymbol{\beta}|<2k} \frac{D_2^{\boldsymbol{\beta}} k(\boldsymbol{x}, \boldsymbol{x})}{\boldsymbol{\beta}!}(\boldsymbol{x}_i - \boldsymbol{x})^{\boldsymbol{\beta}} + R(\boldsymbol{x}, \boldsymbol{x}_i). \tag{2.23}$$

If we insert (2.23) into (2.22) and once more take advantage of the polynomial precision property of the coefficient vector u we are left with

$$Q(\boldsymbol{u}) = -\sum_j u_j \left[R(\boldsymbol{x}, \boldsymbol{x}_j) + R(\boldsymbol{x}_j, \boldsymbol{x}) - \sum_i u_i R(\boldsymbol{x}_i, \boldsymbol{x}_j) \right].$$

Now Theorem 2.3.4 allows us to bound $\sum_j |u_j| \leq c_1$.

Moreover, $||\boldsymbol{x} - \boldsymbol{x}_j||_2 \leq c_2 h_{\mathcal{X},\Omega}$ and $||\boldsymbol{x}_i - \boldsymbol{x}_j||_2 \leq 2c_2 h_{\mathcal{X},\Omega}$.

Therefore, all three remainder terms can be bounded by an expression of the form $C h_{\mathcal{X},\Omega}^{2k} C_k(\boldsymbol{x})$, where we made use the interior cone property of Ω enabling us to define the term $C_k(\boldsymbol{x})$. Combining these bounds and taking the square root yields the stated bound for the power function. □

Theorem 2.3.5 says that interpolant with $\mathcal{C}^{2k}$ smooth kernel k has approximation order k. Thus, for infinitely smooth strictly positive definite functions such as the Gaussians, we see that the approximation order k is arbitrarily high. For strictly positive definite functions with limited smoothness such as the Matérn functions, the approximation order is limited by the smoothness of the basis function.

For functions with finite smoothness it is possible to bound the constant $\mathcal{C}_k(\boldsymbol{x})$ by some additional powers of h, and thereby to improve the order predicted by Theorem 2.3.5.

For instance,for Matérn functions,$\beta > \frac{s}{2}$, we can obtain

$$|D^\alpha f(\boldsymbol{x}) - D^\alpha P_f| \leq C h_{\mathcal{X},\Omega}^{\beta - \frac{s}{2} - |\alpha|} |f|_{\mathcal{N}_k}.$$

provided $|\alpha| \leq \beta - \frac{s+1}{2}$, $h_{\mathcal{X},\Omega}$ is sufficiently small, and $f \in \mathcal{N}_k$.[14]

Chapter 3

Bayesian optimization

3.1 Introduction to Bayesian Optimization

[15] is the main source of this chapter.

In this chapter we are going to define and analyze the most important tool we will use in this work: Bayesian Optimization (BO). See [9] for a more detailed work on Bayesian Optimazion.

Despite its name, BO incorporates frequentist methods to achieve its results. The term "Bayesian" primarily refers to the approach used in modeling the underlying objective function as a probabilistic model, usually a Gaussian process. This model captures the uncertainty in predictions about the function's behavior. However, the optimization process itself relies on frequentist principles.

BO aims to find the optimum of an unknown, possibly non-convex function. Since little information is known about the underlying function f, BO requires estimating a surrogate function to model the unknown function.

Now we fix the setting that we will have from now on.

As in Chapter 2, $\mathcal{H}_k$ represents a separable reproducing kernel Hilbert space with reproducing kernel $k(\cdot,\cdot)$ and $||\cdot||_{\mathcal{H}_k}$ is the norm in $\mathcal{H}_k$. Furthermore, $||\cdot||$ denotes the l_2 norm. Let $B_k = \{f : f \in \mathcal{H}_k, ||f||_{\mathcal{H}_k} \leq B\}$ denote a bounded subset in the RKHS and $X \subset \mathbb{R}^d$ a compact subset in $\mathbb{R}^d$.

Let $f : \mathcal{X} \to \mathbb{R}$ be the unknown black-box function to be optimized, where $\mathcal{X} \subset \mathbb{R}^d$ is a compact set. BO aims to find a maximum $\boldsymbol{x}^*$ of the function f, i.e.,

$$f(\boldsymbol{x}^*) = \max_{\boldsymbol{x} \in \mathcal{X}} f(\boldsymbol{x}).$$

We are going to analyze an Algorithm of sequential BO. In sequential BO, a single point $\boldsymbol{x}_t \in \mathcal{X}$ is selected to query an observation at round t. The effectiveness of a selection policy can be measured by the cumulative regret R_T and simple regret r_T over the T steps. The cumulative regret R_T and

simple regret r_T are defined as follows,

$$R_T = \sum_{t=1}^{T} (f(\boldsymbol{x}^*) - f(\boldsymbol{x}_t)), \tag{3.1}$$

$$r_T = f(\boldsymbol{x}^*) - \max_{1 \leq t \leq T} f(\boldsymbol{x}_t). \tag{3.2}$$

The regret bound introduced in numerous theoretical works is based on the *maximum information gain* defined as

$$\gamma_T = \max_{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_T} \frac{1}{2} \log \det(I_T + \sigma^{-2} K^T). \tag{3.3}$$

The bounds of γ_T for commonly used kernels are studied in (Srinivas et al., 2010). Specifically, Srinivas et al.(2010) state that $\gamma_T = \mathcal{O}(d \log T)$ for the linear kernel, $\gamma_T = \mathcal{O}((\log T)^{d+1})$ for the squared exponential kernel and $\gamma_T = \mathcal{O}(T^{\alpha}(\log T))$ for the Matérn kernels with $\nu > 1$, where $\alpha = \frac{d(d+1)}{2\nu + d(d+1)} \leq 1$.

We employ the term γ_T to build the regret bounds of our algorithms. There are two possible settings: noise-free setting and perturbation setting.

Noise-Free Setting: when it is assumed that the underlying function f belongs to an RKHS associated with kernel $k(\cdot, \cdot)$, i.e., $f \in \mathcal{H}_k$, with $||f||_{\mathcal{H}_k} < \infty$. In the noise-free setting, we can directly observe $f(\boldsymbol{x})$, $\boldsymbol{x} \in X$ without noise perturbation.

Perturbation Setting: In the perturbation setting, we cannot observe the function evaluation $f(\boldsymbol{x})$ directly. Instead, we observe $y = f(\boldsymbol{x}) + g(\boldsymbol{x})$, where $g(\boldsymbol{x})$ is an unknown perturbation function.

In this work, we consider a noise-free perturbation setting.

3.2 BO Sequential Selection in Noise Free Setting

In order to present the first algorithm, we have to define first of all two functions. We define $m_t(\boldsymbol{x})$ and $\sigma_t(\boldsymbol{x})$ as follows:

$$m_t(\boldsymbol{x}) = \boldsymbol{k}_t(\boldsymbol{x})^T K_t^{-1} f_t, \tag{3.4}$$

$$\sigma_t^2(\boldsymbol{x}) = k(\boldsymbol{x}, \boldsymbol{x}) - \boldsymbol{k}_t, (\boldsymbol{x})^T K_t^{-1} \boldsymbol{k}_t(\boldsymbol{x}) \tag{3.5}$$

where $\boldsymbol{k}_t(\boldsymbol{x}) = [k(\boldsymbol{x}, \boldsymbol{x}_1), \ldots, k(\boldsymbol{x}, \boldsymbol{x}_t)]^T$ and the kernel matrix $K_t = [k(\boldsymbol{x}_i, \boldsymbol{x}_j)]_{1 \leq i,j \leq t}$.

Someone could wonder about the meaning of these two functions. They are actually nothing more than the approximating function 2.12 and the power function 2.15, with the difference that they also depend on the number of iterations of the algorithm.

The sequential optimization uses these functions and their combination

$$c(\boldsymbol{x}) = m_{t-1}(\boldsymbol{x}) + ||f||_{\mathcal{H}_k} \sigma_{t-1}^2(\boldsymbol{x}). \tag{3.6}$$

Algorithm 1

for t=1 **to** T **do**
Obtain $m_{t-1}(\cdot)$ and $\sigma^2_{t-1}(\cdot)$ via equations (3.4) and (3.5)
Choose $x_t = \text{argmax}_{x\in\mathcal{X}} c(x)$
end for

Now we describe the Algorithm 1 that will be adopted to our application.

The regret bounds of the Algorithm 1 are given in Theorem 3.2.1.

Theorem 3.2.1. *Suppose $f \in \mathcal{H}_k$ associated with $k(\boldsymbol{x},\boldsymbol{x}) \leq 1$, $||f||_{\mathcal{H}_k} < \infty$. Let $C_1 = \frac{8}{\log(1+\sigma^{-2})}$. Algorithm 1 achieves a cumulative regret bound and a simple regret bound given as follows:*

$$R_T \leq ||f||_{\mathcal{H}_k}\sqrt{TC_1\gamma_T}$$
$$r_T \leq ||f||_{\mathcal{H}_k}\sqrt{\frac{C_1\gamma_T}{T}}$$

dove $0 < c < +\infty$.

The regret that for us is more important is the simple regret, which describes the deviation from the maximum. This theorem is useful because it tells that when we increase the number of iterations, the simple regret bound decreases.

In order to prove this theorem we use the following lemmas:

Lemma 3.2.1. *Suppose $f \in \mathcal{H}_k$ associated with $k(\boldsymbol{x},\boldsymbol{x})$, then*

$$(m_t(\boldsymbol{x}) - f(\boldsymbol{x}))^2 \leq ||f||^2_{\mathcal{H}_k}\sigma^2_t(\boldsymbol{x}).$$

Proof. Let $\boldsymbol{\alpha} = K_t^{-1}\boldsymbol{k}_t(\boldsymbol{x})$. Then we have

$$\begin{aligned}
(m_t(\boldsymbol{x}) - f(\boldsymbol{x}))^2 &= \left(\sum_{i=1}^{t}\alpha_i f(\boldsymbol{x}_i) - f(\boldsymbol{x})\right)^2 \\
&= \left(\left\langle \sum_{i=1}\alpha_i k(\cdot,\boldsymbol{x}_i) - k(\cdot,\boldsymbol{x}), f\right\rangle_{\mathcal{H}_k}\right)^2 \\
&\leq \langle f,f\rangle_{\mathcal{H}_k}\left\langle \sum_{i=1}^{t}\alpha_i k(\cdot,\boldsymbol{x}_i) - k(\cdot,\boldsymbol{x}), \sum_{i=1}^{t}\alpha_i k(\cdot,\boldsymbol{x}_i) - k(\cdot,\boldsymbol{x})\right\rangle_{\mathcal{H}_k} \\
&= ||f||^2_{\mathcal{H}_k}||\sum_{i=1}^{t}\alpha_i k(\cdot,\boldsymbol{x}_i) - k(\cdot,\boldsymbol{x})||^2_{\mathcal{H}_k}
\end{aligned}$$

In addition, we can achieve that

$$\begin{aligned}
||\sum_{i=1}^{t}\alpha_i k(\cdot,\boldsymbol{x}_i) - k(\cdot,\boldsymbol{x})||^2_{\mathcal{H}_k} &= k(\boldsymbol{x},\boldsymbol{x}) - 2\sum_{i=1}^{t}\alpha_i k(\boldsymbol{x}_i,\boldsymbol{x}) \\
&\quad + \sum_{i=1}^{t}\sum_{j=1}^{t}\alpha_i k(\boldsymbol{x}_i,\boldsymbol{x})\alpha_j k(\boldsymbol{x}_j,\boldsymbol{x}) \\
&= k(\boldsymbol{x},\boldsymbol{x}) - 2\boldsymbol{\alpha}^T k_t(\boldsymbol{x}) + \boldsymbol{\alpha}^T K_t \boldsymbol{\alpha} \\
&= k(\boldsymbol{x},\boldsymbol{x}) - 2\boldsymbol{k}_t(\boldsymbol{x})^T K_t^{-1}\boldsymbol{k}_t(\boldsymbol{x}) \\
&\quad + \boldsymbol{k}_t^T(\boldsymbol{x})K_t^{-1}K_t K_t^{-1}\boldsymbol{k}_t(\boldsymbol{x}) \\
&= k(\boldsymbol{x},\boldsymbol{x}) - \boldsymbol{k}_t(\boldsymbol{x})^T K_t^{-1}\boldsymbol{k}_t(\boldsymbol{x}) \\
&= \sigma_t^2(\boldsymbol{x})
\end{aligned}$$

Plugging this last equality in the previous one, we can attain

$$(m_t(\boldsymbol{x}) - f(\boldsymbol{x}))^2 \leq ||f||^2_{\mathcal{H}_k}\sigma_t^2(\boldsymbol{x}). \tag{3.7}$$

□

This Lemma suggests that the pointwise squared difference between f and m_t is bounded by the value of σ_t^2 at that point.

Furthermore, the inequality (3.7) is precisely the analogue of the inequality (2.16), except that it represents the squared error of the algorithm at step t and not in a setting with a fixed set of nodes as in Chapter 2.

Lemma 3.2.2. *We have that*

$$f(\boldsymbol{x}^*) - f(\boldsymbol{x}_t) \leq 2||f||_{\mathcal{H}_k}\sigma_{t-1}(\boldsymbol{x}_t).$$

Proof. From Lemma 3.2.1 we attain:

$$f(\boldsymbol{x}) \leq m_t(\boldsymbol{x}) + ||f||_{\mathcal{H}_k}\sigma_t(\boldsymbol{x}) \quad \forall \boldsymbol{x}. \tag{3.8}$$

Therefore, we can consider the inequality at the iteration $t-1$ and for the hypothetical maximum x^*. We achieve that

$$f(\boldsymbol{x}^*) \leq m_t(\boldsymbol{x}^*) + ||f||_{\mathcal{H}_k}\sigma_t(\boldsymbol{x}^*). \tag{3.9}$$

Finally, putting all together and using the fact that x_t is the maximum at round $t-1$ for the function (3.6), we can obtain the thesis

$$\begin{aligned}
f(\boldsymbol{x}^*) - f(\boldsymbol{x}_t) &\leq m_{t-1}(\boldsymbol{x}^*) + ||f||_{\mathcal{H}_k}\sigma_{t-1}(\boldsymbol{x}^*) - f(\boldsymbol{x}_t) \\
&\leq m_{t-1}(\boldsymbol{x}_t) + ||f||_{\mathcal{H}_k}\sigma_{t-1}(\boldsymbol{x}_t) - f(\boldsymbol{x}_t) \\
&\leq m_{t-1}(\boldsymbol{x}_t) + ||f||_{\mathcal{H}_k}\sigma_{t-1}(\boldsymbol{x}_t) - m_{t-1}(\boldsymbol{x}_t) + ||f||_{\mathcal{H}_k}\sigma_{t-1}(\boldsymbol{x}_t) \\
&\leq ||f||_{H_k}\sigma_{t-1}(\boldsymbol{x}_t) + ||f||_{\mathcal{H}_k}\sigma_{t-1}(\boldsymbol{x}_t) \\
&= 2||f||_{\mathcal{H}_k}\sigma_{t-1}(\boldsymbol{x}_t).
\end{aligned}$$

□

Lemma 3.2.3. *Let* $\hat{\sigma}_t^2 = k(\boldsymbol{x}, \boldsymbol{x}) - \boldsymbol{k}_t(\boldsymbol{x})^T(\sigma^2 I + K_t)^{-1}\boldsymbol{k}_t(\boldsymbol{x})$. *Then*

$$\sigma_t^2(\boldsymbol{x}) \leq \hat{\sigma}_t^2(\boldsymbol{x}).$$

Proof. Since kernel matrix K_t is positive semi-definite, we can write it in the following way,

$$K_t = U^T \Lambda U,$$

where U is orthonormal matrix which consists of eigenvectors, Λ is a diagonal matrix which consists of eigenvalues.

Let $\boldsymbol{\beta} = U\boldsymbol{k}_t(\boldsymbol{x})$, then we obtain

$$\begin{aligned}
\boldsymbol{k}_t(\boldsymbol{x})^T(\sigma^2 I + K_t)^{-1}\boldsymbol{k}_t(\boldsymbol{x}) &= \boldsymbol{k}_t(\boldsymbol{x})^T(\sigma^2 U^T U + U^T \Lambda U)^{-1}\boldsymbol{k}_t(\boldsymbol{x}) \\
&= \boldsymbol{k}_t(\boldsymbol{x})^T[U^T(\sigma^2 I + \Lambda)U]^{-1}\boldsymbol{k}_t(\boldsymbol{x}) \\
&= \boldsymbol{\beta}^T(\sigma^2 I + \Lambda)^{-1}\boldsymbol{\beta}.
\end{aligned}$$

Therefore, using the previous chain of equality, we can deduce the inequality

$$\begin{aligned}
\boldsymbol{k}_t(\boldsymbol{x})^T(\sigma^2 I + K_t)^{-1}\boldsymbol{k}_t(\boldsymbol{x}) &= \sum_{i=1}^{t} \frac{\beta_i^2}{\sigma^2 + \lambda_i} \\
&\leq \sum_{i=1}^{t} \frac{\beta_i^2}{\lambda_i} = \beta^T \Lambda^{-1} \beta \\
&= \boldsymbol{k}_t(\boldsymbol{x}) U^T \Lambda^{-1} U \boldsymbol{k}_t(\boldsymbol{x}) \\
&= \boldsymbol{k}_t(\boldsymbol{x})^T K_t^{-1} \boldsymbol{k}_t(\boldsymbol{x})
\end{aligned}$$

It follows that

$$\begin{aligned}
\sigma_t^2(\boldsymbol{x}) &= k(\boldsymbol{x}, \boldsymbol{x}) - \boldsymbol{k}_t(\boldsymbol{x})^T K_t^{-1} \boldsymbol{k}_t(\boldsymbol{x}) \\
&\leq k(\boldsymbol{x}, \boldsymbol{x}) - \boldsymbol{k}_t(\boldsymbol{x})^T(\sigma^2 I + K_t)^{-1}\boldsymbol{k}_t(\boldsymbol{x}) \\
&= \hat{\sigma}^2(\boldsymbol{x})
\end{aligned}$$

□

Now we are ready for the proof of the Theorem 3.2.1.

Proof. First, we have

$$\begin{aligned}
R_T &= \sum_{t=1}^{T} (f(\boldsymbol{x}_*) - f(\boldsymbol{x}_t)) \\
&\leq 2||f||_{\mathcal{H}_k} \sum_{t=1}^{T} \sigma_{t-1}(\boldsymbol{x}_t) \\
&\leq 2||f||_{\mathcal{H}_k} \sqrt{T \sum_{t=1}^{T} \sigma_{t-1}^2(\boldsymbol{x}_t)},
\end{aligned}$$

where we have used Lemma 3.2.2 and a fundamental inequality.

Since

$$s \leq \frac{1}{\log(1+\sigma^{-2})} \log(1+\sigma^{-2}s) \quad \text{for} \quad s \in [0,1]$$

and

$$0 \leq \hat{\sigma}_{t-1}(\boldsymbol{x}_t) \leq k(\boldsymbol{x},\boldsymbol{x}) \leq 1,$$

it follows that

$$\begin{aligned}\sum_{t=1}^{T} \sigma_{t-1}^2(\boldsymbol{x}_t) \leq \sum_{t=1}^{T} \hat{\sigma}_{t-1}^2(\boldsymbol{x}_t) &\leq \frac{1}{\log(1+\sigma^{-2})} \sum_{i=1}^{T} \log(1+\sigma^{-2}\hat{\sigma}_{t-1}^2(\boldsymbol{x}_t)) \\ &\leq \frac{2\gamma_T}{\log(1+\sigma^{-2})}.\end{aligned}$$

Now we attain that

$$\begin{aligned}R_T &\leq 2||f||_{\mathcal{H}_k} \sqrt{\frac{2T\gamma_T}{\log(1+\sigma^{-2})}} \\ &= ||f||_{\mathcal{H}_k} \sqrt{TC_1\gamma_T}\end{aligned}$$

It follows that $r_T \leq \frac{R_T}{T} \leq ||f||_{\mathcal{H}_k} \sqrt{\frac{C_1\gamma_T}{T}}$.

□

3.3 Robust Initialization for BO

The initialization phase of BO is important. In [15], it is discussed how to achieve robust initialization, which means the process of setting the initial conditions of an algorithm or system in a manner that ensures reliable performance and convergence, even in the presence of uncertainties, variations, or noise in the data or environment.

In [15] it is shown that algorithms that attain a small fill distance can achieve small adversarial regret bounds.

We recall the definition of fill distance and separate distance of a set of points $X = \{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_T\}$:

$$h_X = \sup_{\boldsymbol{x} \in \mathcal{X}} \min_{\boldsymbol{x}_t \in X} ||\boldsymbol{x} - \boldsymbol{x}_t||,$$

$$\rho_X = \frac{1}{2} \min_{\boldsymbol{x}_i, \boldsymbol{x}_j \in X, \boldsymbol{x}_i \neq \boldsymbol{x}_j} ||\boldsymbol{x}_i - \boldsymbol{x}_j||.$$

Our previous algorithm benefits from an initialization set X_{T-1} which has the aim of minimizing the fill distance.

The following theorem theorem ensures what we have just said.

Theorem 3.3.1. *Define* $\mathcal{B}_k = \{f : f \in \mathcal{H}_k, ||f||_{\mathcal{H}_k} \leq B\}$ *associated with* $k(\boldsymbol{x}, \boldsymbol{x})$ *for* $\boldsymbol{x} \in \mathcal{X} \subset \mathbb{R}^d$. *Suppose* $f \in \mathcal{B}_k$ *and* $\mathcal{H}_k$ *is norm-equivalent to the Sobolev space of order* s. *Then there exists a constant* $C > 0$, *such that the query point set generated by the Algorithm, with a sufficiently small covering radius (fill distance)* h_X *achieves a regret bound given by :*

$$\tilde{r}_T \leq BCh_X^{s-d/2}. \tag{3.10}$$

Observation 3.3.1. We can notice that the regret bound decreases as the covering radius becomes smaller. This means that a query set with a small covering radius can guarantee a small regret.

Now we consider two functions and we apply our BO, in order to verify the effectiveness of this tool.

Example 3.3.1. The first function that we consider is

$$f : [-1, 1]^2 \to \mathbb{R},\ (x, y) \mapsto f(x, y) := \frac{1}{(x-a)^2 + (y-b)^2 + 3}$$

where $a = 0.57$ and $b = -0.14$. In this case, the existence of a maximum can be easily proved as a simple exercise of analysis, and it can also be observed from its geometrical representation in Figure 3.1.

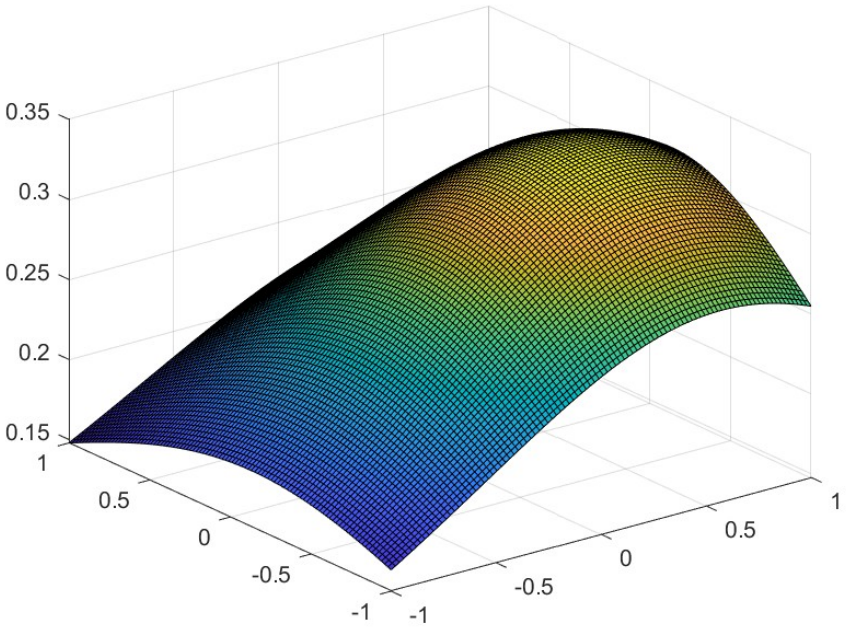
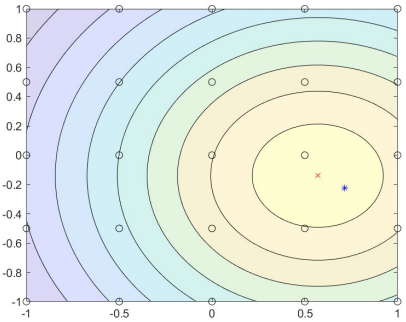


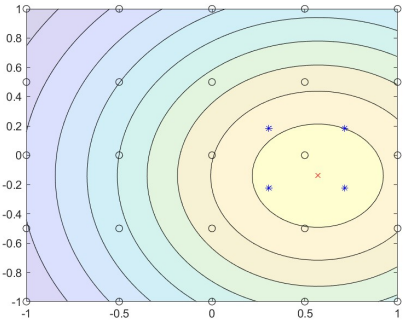
Figure 3.1: Plot of $f(x, y) = \frac{1}{(x-a)^2+(y-b)^2+3}$.

We apply BO Algorithm 1 to this function with Matérn $\mathcal{C}^0$ kernel.

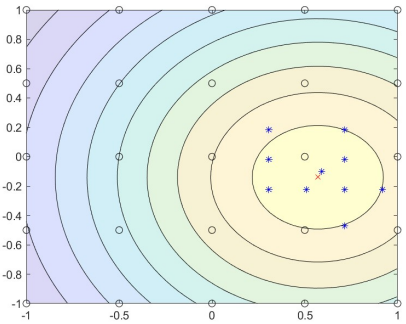
In Figure 3.2 we present the results of our algorithm at the first iteration, the fourth iteration, the tenth iteration and the thirtieth iteration, where the real maximum is indicated by a red cross. The initial set of nodes is a gridded set of 25 points, which are displayed as empty circles.



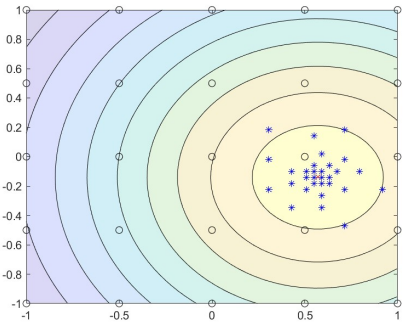
(a) First iteration BO.



(b) Fourth iteration BO.



(c) Tenth iteration BO.



(d) Thirtieth iteration BO.

Figure 3.2: Performance of the BO algorithm.

Initially, we can see that the algorithm moves away from the maximum and then stabilizes around it at the thirtieth iteration.

Let us proceed with the analysis of the cumulative regret and of the simple regret. We plot the two regrets and we obtain Figure 3.3. On the horizontal axis we have the number of iterations used in the algorithm, while, on the vertical axis, the values that the functions take on the point visited during the corresponding iteration.

The blue line represents the cumulative regret R_T. From its definition (3.1), it can be seen that it is an increasing quantity with respect to the number of iterations. and this is consistent with our plot. The red line, on

the other hand, represents the simple regret r_T. Also in this case, from its definition (3.2), we can observe that is a decreasing function, and again the graph is consistent with theory.

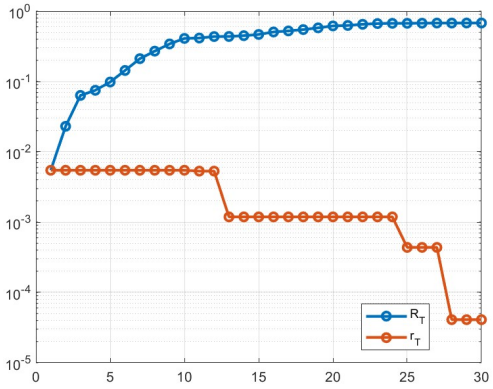


Figure 3.3: Comparison between cumulative regret and simple regret.

Furthermore, we show the trends of the approximant and the objective function as iterations vary.

As the number of iterations increases, we have that the surrogate function begins to have behavior similar to the objective function.

In conclusion, we can see that in not many iterations the algorithm leads to a value which is near the maximum, because the estimated maximum is $(0.550, -0.143)$, while the real one is $(0.57, -0.14)$.

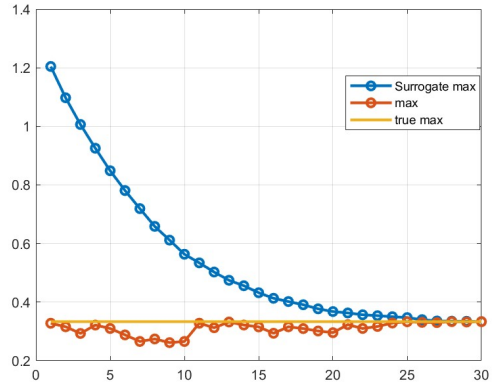


Figure 3.4: Comparison between surrogate model and objective function.

In the following example, we are going to consider a function that is more

complicated to deal with, in order to see if our algorithm can be effective also in this case.

Example 3.3.2. The second function that we consider is

$$f : [-1,1]^2 \to \mathbb{R}$$
$$(x,y) \mapsto f(x,y) := \frac{\cos\left(20\frac{(x-a)^2+(y-b)^2}{(x-a)^2+(y-b)^2+1}\right)}{(x-a)^2+(y-b)^2+3},$$

where $a = 0.57$ and $b = -0.14$.

In this case, the research of a maximum is clearly more complicated, as we can see from the geometrical representation in Figure 3.5.

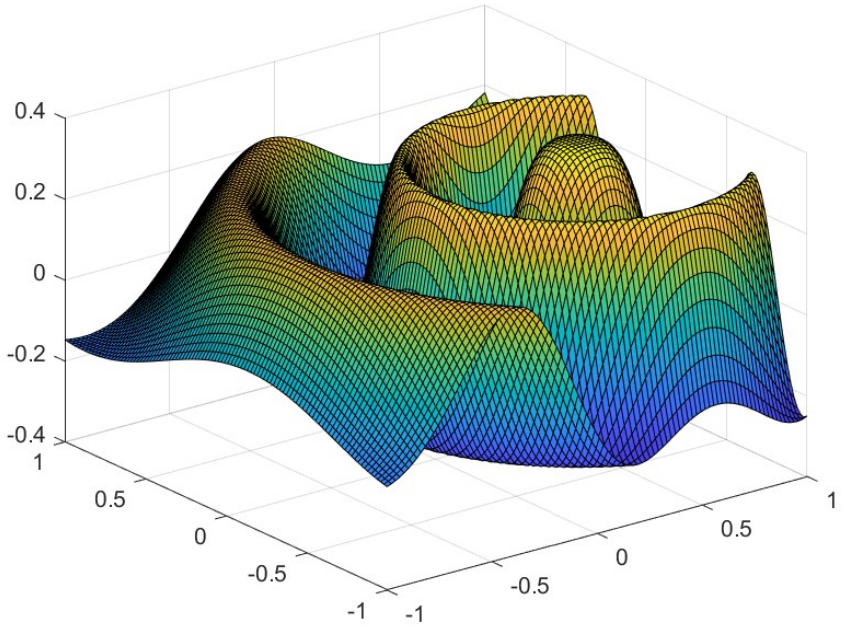
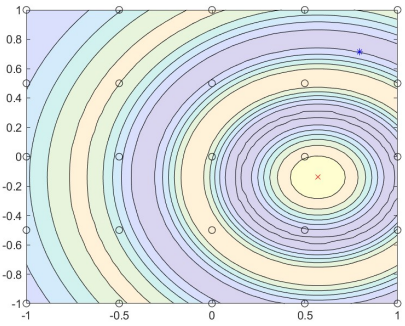


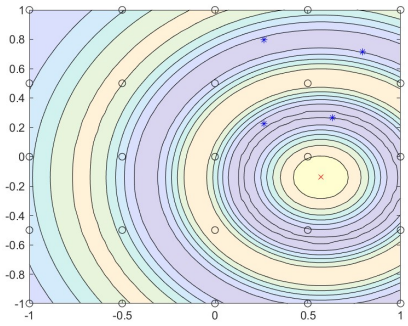
Figure 3.5: Geometrical representation of $f(x,y) = \frac{\cos\left(20\frac{(x-a)^2+(y-b)^2}{(x-a)^2+(y-b)^2+1}\right)}{(x-a)^2+(y-b)^2+3}$.

In Figure 3.6, we present the results of BO Algorithm 1 with Gaussian kernel (shape parameter 2) at the first iteration, the fourth iteration, the tenth iteration and the thirtieth iteration. The starting grid is formed by 25 equispaced points. We can observe that the algorithm leaves the region in which we have the real maximum (marked with the red cross) and finally it concentrates on the ridges with higher values.

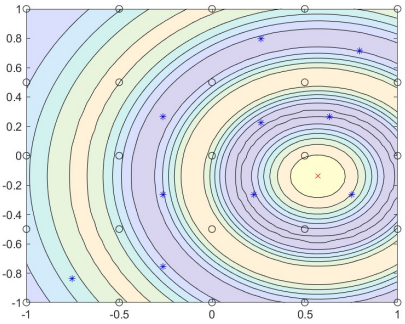
However, we notice that it does not converge to the real maximum in this experiment.



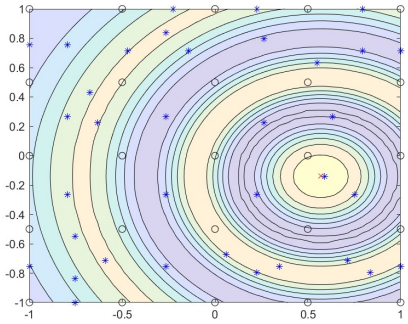
(a) First iteration BO.



(b) Fourth iteration BO.



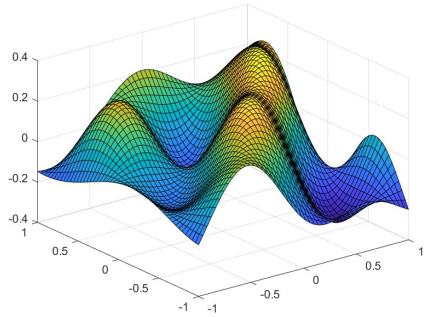
(c) Tenth iteration BO.



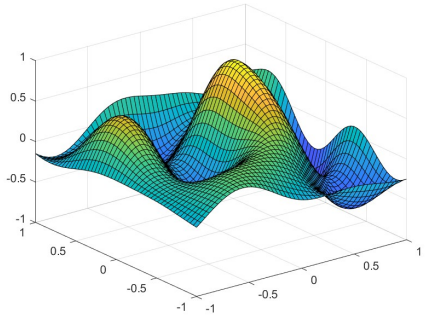
(d) Thirtieth iteration BO.

Figure 3.6: Performance of the Bayesian Optimization Gaussian kernel on uniform starting initial grid.

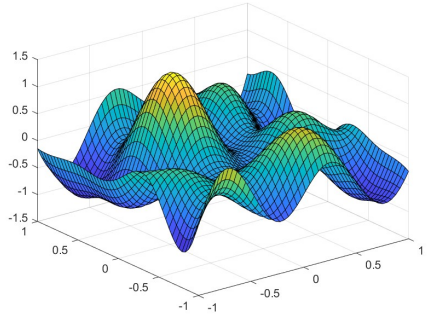
The reason why the algorithm at the beginning moves away for some iteration is that, as we can see in Figure 3.7, for those iterations, the surrogate model presents there its maximum. However, when the number of iteration increases we know from the theory that the surrogate model tends more and more to the real function, and thus we have the reduction of the error.



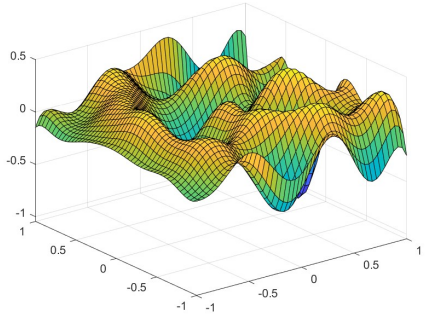
(a) Surrogate model at first iteration (uniform starting grid).



(b) Surrogate model at fourth iteration (uniform starting grid).



(c) Surrogate model at tenth iteration (uniform starting grid).



(d) Surrogate model at tenth iteration (uniform starting grid).

Figure 3.7: Evolution of the surrogate model with uniform starting grid.

Obviously, we can observe also in this case the comparison between the surrogate function and the objective function on the visited nodes in Figure 3.9 and the comparison between the cumulative regret and the simple regret in Figure 3.8.

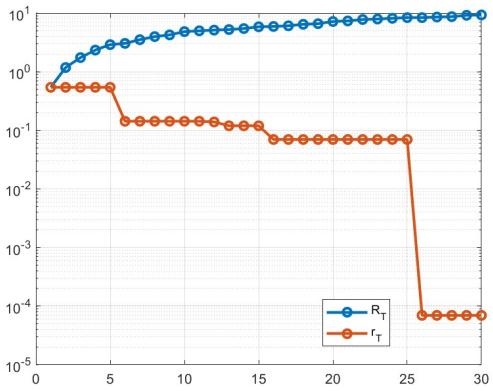


Figure 3.8: Comparison between cumulative regret and simple regret.

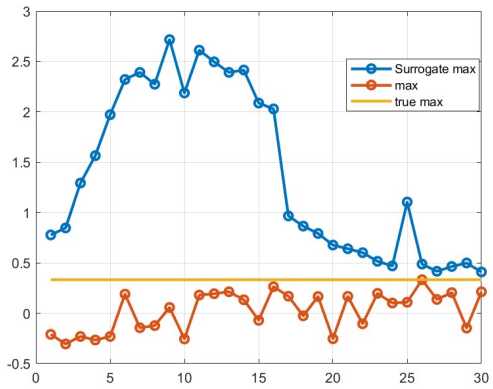
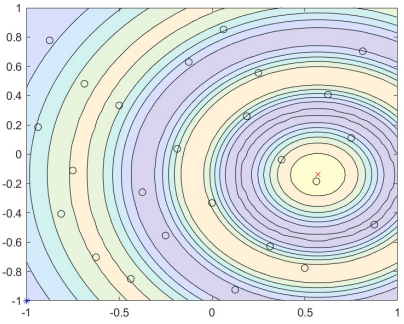


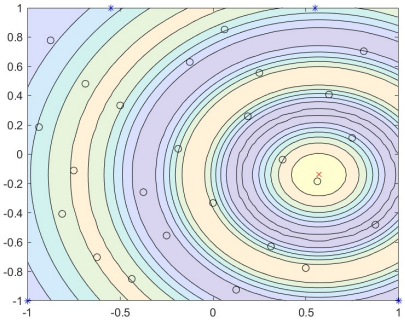
Figure 3.9: Comparison between surrogate model and objective function.

Now we pass from a uniform starting grid to a non-uniform starting grid. In Figure 3.10 we can observe a different position of the starting nodes from that of Figure 3.6.

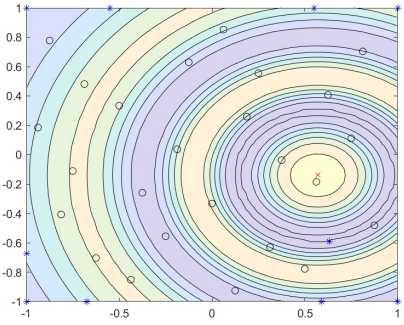
As far as the second experiment is concerned, at the tenth iteration we are still far from the areas where the highest values are present, which is not the case of the first experiment as we can see in Figure 3.6 .



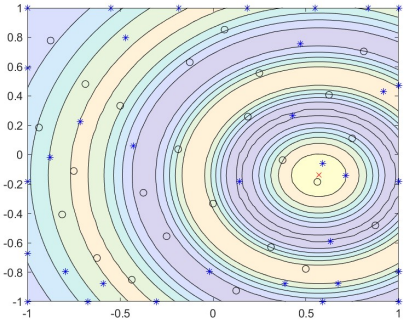
(a) First iteration BO.



(b) Fourth iteration BO.

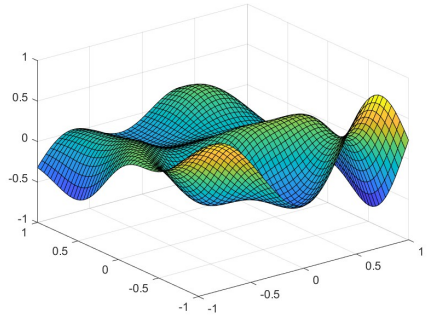


(c) Tenth iteration BO.

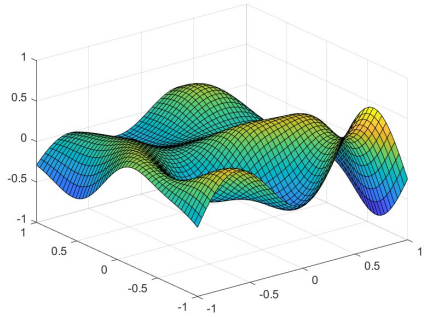


(d) Thirtieth iteraton BO.

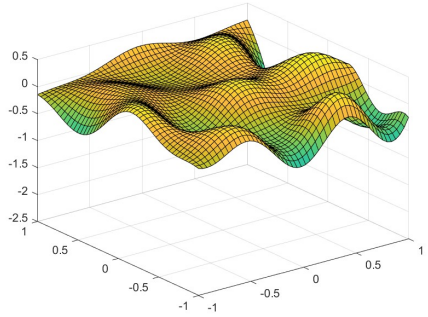
Figure 3.10: Performance of the Bayesian Optimization Gaussian kernel with initial non-uniform-grid.



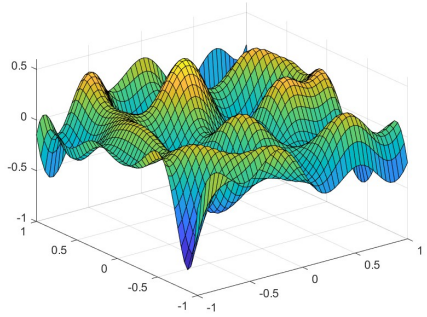
(a) Surrogate model at first iteration (non-uniform starting grid).



(b) Surrogate model at fourth iteration (non-uniform starting grid).



(c) Surrogate model at tenth iteration (non-uniform starting grid).



(d) Surrogate model at thirtieth iteration (non-uniform starting grid).

Figure 3.11: Evolution of the surrogate model with non-uniform starting grid.

Furthermore, comparing the trend of the surrogate maximum calculated on the uniform grid in Figure 3.9 versus the one calculated on the non-uniform grid 3.12, we realize that the latter has more instability, at 25th iteration, in fact, we notice an approximation error that can also be seen from the geometric representation of the surrogate model in Figure 3.13.

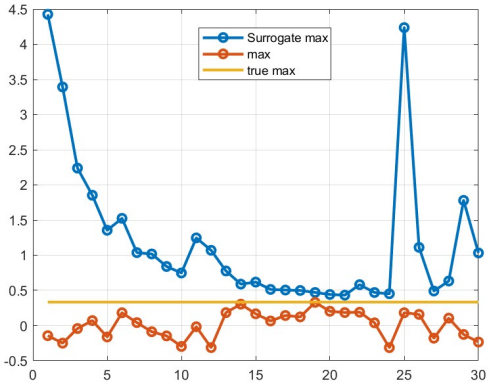


Figure 3.12: Comparison between surrogate model on a non-uniform starting grid and objective function.

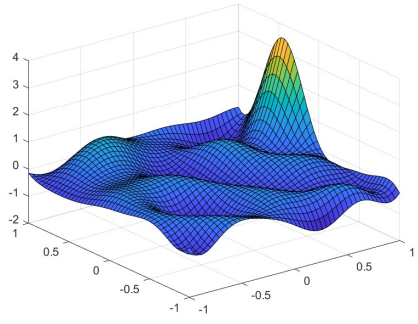
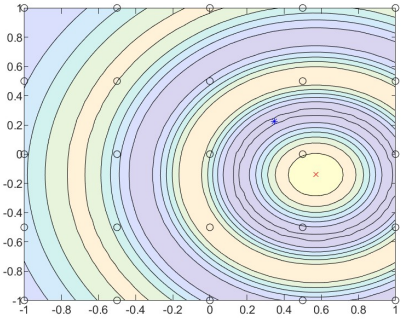


Figure 3.13: Geometrical representation of the surrogate model on a non-uniform starting grid at the 25th iteration.

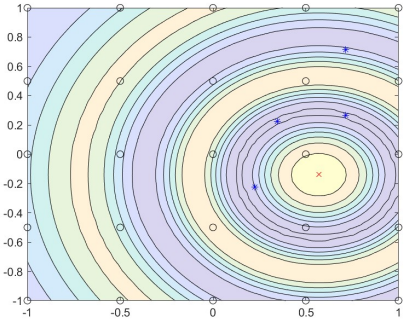
Therefore, from the empirical experience, it is evident that Bayesian Optimization (BO) performs better on a uniform grid compared to a non-uniform grid. This finding highlights the significance of grid structure in optimizing the efficiency and effectiveness of BO.

This is also consistent with theory because, for the same number of nodes, the uniform grid is the configuration which minimizes fill distance and maximizes separation distance and this leads to a smaller simple regret, as we have said in Theorem 3.3.1.

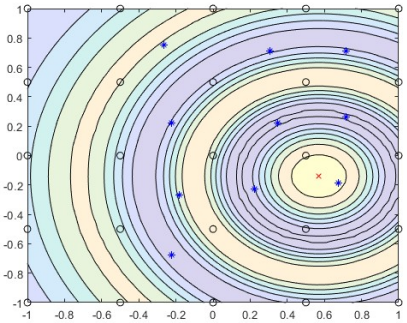
However, as can be seen from these two experiments, at the 30th iteration we still do not have stability of the method. Let us try to see what happens by changing the shape parameter. As we can see from Figure 3.14, we can manage to attain a better convergence.



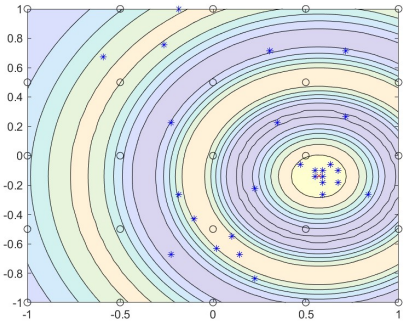
(a) First iteration BO.



(b) Fourth iteration BO.



(c) Tenth iteration BO.



(d) Thirtieth iteraton BO.

Figure 3.14: Performance of the Bayesian Optimization Gaussian kernel with shape parameter 10.

In conclusion, we have empirically shown that using a uniform grid and tuning carefully the shape parameter can enhance the convergence of Bayesian Optimization, leading to more efficient and accurate results.

Chapter 4

Image similarity measures

Trying to estimate the similarity of two images is a complex task in the image processing community. Digital images are usually subject to a wide variety of distortions during acquisition, processing, compression, storage, transmission and reproduction, any of this may result in a degradation of visual quality.

Usually when we want to quantify visual image quality, we do a subjective assessment. However, in practice, a subjective assessment is inconvenient, time-consuming and expensive.

A goal of research in objective image evaluation is to develop quantitative measures that can automatically relate to image quality.

Objective quality metrics can be ranked according to the availability of a not distorted original image :

1. **Full-reference metrics:** if we know the original image;
2. **No-reference or "blind" quality assessment metrics:** if we do not have the original image.

Now we present some of these metrics that will be used in our application in Chapter 6.[16]

4.1 MSE: Mean Squared Error

First of all, we talk about the most common estimator of image quality, the *mean squared error* (MSE).

It is a full reference metric because it needs the real signal in order to be computed. Let us see the formal definition.

Definition 4.1.1. Let $f, \hat{f} : \Omega \to \mathbb{R}$ be two functions which represent the reference and the distorted image respectively, Ω a compact subset of $\mathbb{R}^2$.

The *mean squared error* between f and $\hat{f}$ is defined as

$$MSE := \frac{1}{MN} \sum_{n=1}^{M} \sum_{m=1}^{N} [\hat{f}(x,y) - f(x,y)]^2.$$

So MSE is calculated by averaging the squared intensity differences between the pixels. It is certainly simple to calculate, it has a clear physical meaning and is great for mathematical optimization.

4.2 SSIM: Structural Similarity Index

In the case of full-reference approaches, as we have just said, the simplest and most widely used metric is the mean squared error (MSE).

However, in the last three decades, MSE has been a bit sidelined because the global interest has gone into the development of quality assessment methods that take advantage of known characteristics of the human visual system (HVS).[16]

In this section, we describe a new metric for quality assessment based on the hypothesis that the HSV is highly adapted for extracting structural information.

Usually, the main characteristics of an image are *luminance* and *contrast*.

Luminance

Typically, *standard luminance* refers to the absolute amount of light emitted by an object per unit area (pixel). Numerically speaking, when we talk about standard luminance of an image, we mean the number associated to the pixel.

In this work, in order to be consistent with the metric used in [11], by *luminance*, we mean the average of the standard luminances.

Therefore, assuming $\boldsymbol{x}$ a signal modelled by a simple function, x_i the standard luminance associated to i-th pixel, the luminance is estimated as the mean intensity of the pixels:

$$\mu_{\boldsymbol{x}} = \frac{1}{N} \sum_{i=1}^{N} x_i. \tag{4.1}$$

Contrast

The other fundamental characteristic of images is *contrast*. Contrast refers to the amount of luminance or grayscale differentiation that exists between various image features. There are more than one definition of contrast.

For instance, Weber contrast (WC) is calculated as the difference in luminance between a feature and background:

$$WC := \frac{I - I_b}{I_b},$$

with I and I_b representing the luminance of the features and the background, respectively.

In this work, the contrast is defined as the sum of the squared differences between the standard luminance of the pixel and the luminance of the signal.

Practically, it is the standard deviation

$$\sigma_{\boldsymbol{x}} = \left(\frac{1}{N-1} \sum_{i=1}^{N} (x_i - \mu_{\boldsymbol{x}})^2 \right).$$

Finally, we consider an element which represents the features of HSV: the *structure* of an image.

Structure

With structure we mean the characteristics of the signal that are independent from the contrast and luminance. So we take a signal $\boldsymbol{x}$ and we compute the normalization in order to study the structure:

$$\frac{\boldsymbol{x} - \mu_{\boldsymbol{x}}}{\sigma_{\boldsymbol{x}}}.$$

Now we want to compare the images for luminance, contrast and structure.

Let $\boldsymbol{x}$ and $\boldsymbol{y}$ be two signals, i.e., two images.

Definition 4.2.1. The *luminance comparison function* is defined as:

$$l(\boldsymbol{x}, \boldsymbol{y}) := \frac{2\mu_{\boldsymbol{x}}\mu_{\boldsymbol{y}} + C_1}{\mu_{\boldsymbol{x}}^2 + \mu_{\boldsymbol{y}}^2 + C_1}, \tag{4.2}$$

where the constant C_1 is included to avoid instability when $\mu_{\boldsymbol{x}}^2 + \mu_{\boldsymbol{y}}^2$ is near to zero. In particular we choose

$$C_1 = (K_1 L)^2,$$

where L is a dynamic range of the pixel values, and $K << 1$ is a small constant.

Symmetrically,

Definition 4.2.2. The *contrast comparison function* is defined as:

$$c(\boldsymbol{x}, \boldsymbol{y}) := \frac{2\sigma_{\boldsymbol{x}}\sigma_{\boldsymbol{y}} + C_2}{\sigma_{\boldsymbol{x}}^2 + \sigma_{\boldsymbol{y}}^2 + C_2} \tag{4.3}$$

where, as in the previous case, the constant C_2 is included to avoid instability when $\sigma_{\boldsymbol{x}}^2 + \sigma_{\boldsymbol{y}}^2$ is near to zero. In particular we choose

$$C_2 = (K_2 L)^2$$

where L is a dynamic range of the pixel values, and $K << 1$ is a small constant.

The structure comparison $s(\boldsymbol{x}, \boldsymbol{y})$ is conducted on the normalized signals $(\boldsymbol{x} - \mu_{\boldsymbol{x}})/\sigma_{\boldsymbol{x}}$ and $(\boldsymbol{y} - \mu_{\boldsymbol{y}})/\sigma_{\boldsymbol{y}}$.

The correlation between these is a simple and effective measure to quantify the structural similarity.

Notice that the correlation between $(\boldsymbol{x}-\mu_{\boldsymbol{x}})/\sigma_{\boldsymbol{x}}$, $(\boldsymbol{y}-\mu_{\boldsymbol{y}})/\sigma_{\boldsymbol{y}}$ is equivalent to the correlation coefficient between $\boldsymbol{x}$ and $\boldsymbol{y}$. Thus, we define the *structure comparison function* as follows:

$$s(\boldsymbol{x}, \boldsymbol{y}) = \frac{\sigma_{\boldsymbol{xy}} + C_3}{\sigma_{\boldsymbol{x}}\sigma_{\boldsymbol{y}} + C_3}.$$

In discrete form, $\sigma_{\boldsymbol{xy}}$ can be estimated as:

$$\sigma_{\boldsymbol{xy}} = \frac{1}{N-1}\sum_{i=1}^{N}(x_i - \mu_{\boldsymbol{x}})(y_i - \mu_{\boldsymbol{y}}).$$

Finally, we combine the three comparisons equations and name the resulting similarity measure the *Structural SIMilarity index* between signals $\boldsymbol{x}$ and $\boldsymbol{y}$:

$$SSIM(\boldsymbol{x}, \boldsymbol{y}) = [l(\boldsymbol{x}, \boldsymbol{y})]^{\alpha}[c(\boldsymbol{x}, \boldsymbol{y})]^{\beta}[s(\boldsymbol{x}, \boldsymbol{y})]^{\gamma},$$

where $\alpha > 0$, $\beta > 0$, $\gamma > 0$ are parameters used to adjust the relative importance of three components.

It is easy to verify that this definition satisfies the three properties:

1. **Symmetry:** $S(\boldsymbol{x}, \boldsymbol{y}) = S(\boldsymbol{y}, \boldsymbol{x})$;
2. **Boundedness:** $S(\boldsymbol{x}, \boldsymbol{y}) \leq 1$;
3. **Unique maximum:** $S(\boldsymbol{x}, \boldsymbol{y}) = 1$ if and only if $\boldsymbol{x} = \boldsymbol{y}$ (in discrete representation, $x_i = y_i$ for all $i = 1, 2, \ldots, N$);

In order to simplify the expression, we set $\alpha = \beta = \gamma = 1$ and $C_3 = C_2/2$ in this paper.

This results in a specific form of the SSIM index:

$$SSIM(\boldsymbol{x}, \boldsymbol{y}) = \frac{(2\mu_{\boldsymbol{x}}\mu_{\boldsymbol{y}} + C_1)(2\sigma_{\boldsymbol{xy}} + C_2)}{(\mu_{\boldsymbol{x}}^2 + \mu_{\boldsymbol{y}}^2 + C_1)(\sigma_{\boldsymbol{x}}^2 + \sigma_{\boldsymbol{y}}^2 + C_2)}.$$

Observation 4.2.1. We have to notice how the comparison functions are defined, so how it is defined the SSIM index, and give an explanation to it. All of them are defined with the ratio $\frac{2xy}{x^2+y^2}$, which can be interpreted in terms of similarity, now we give a geometric interpretation of it.

We consider two vectors $\boldsymbol{u}$ and $\boldsymbol{v}$. The cosine of the angle θ between $\boldsymbol{u}$ and $\boldsymbol{v}$ is given by:

$$\cos(\theta) = \frac{\boldsymbol{u} \cdot \boldsymbol{v}}{|\boldsymbol{u}||\boldsymbol{v}|}.$$

The cosine of the angle is then a measure of their similarity because, when $\cos(\theta) = 1$, the vectors are perfectly aligned (same direction). Furthermore, if $\cos(\theta) = 0$, the vectors are orthogonal and so they do not overlap.

Therefore, more the ratio is near to 1 more the two vectors are "similar".

Now we consider the vectors $\boldsymbol{u} = (x, y)$ and $\boldsymbol{v} = (y, x)$, the ratio obtained from their cosine is

$$\cos(\theta) = \frac{2xy}{x^2 + y^2} \tag{4.4}$$

and describes, for what we have just said, the similarity with respect to the other.

The equality (4.4) invokes precisely the definitions (4.2) and (4.3).

This explains why the comparison functions are defined in that way and why if they assume a value near to 1 the images are similar according with that feature.

4.3 Total variation likelihood measure

Let $f : \Omega \to \mathbb{R}$ be a function which represents a signal, Ω a compact subset of $\mathbb{R}^2$ and $f(\boldsymbol{x})$ the intensity of a pixel $\boldsymbol{x} \in \Omega$.

In section 1.3, we have shown that the phase is a key element of the geometry of an image. In particular, in Example 1.3.1, we have seen that the Total Variation of a signal measures indirectly how its phase is disturbed and therefore how the image is distorted.

In the article [11], it is presented a likelihood measure that uses Total Variation to measure the phase modification as we have just said.

First of all, we recall the definition of Total Variation.

As we have just recalled in Chapter 1, the Total Variation of an image f is

$$TV(f) = \sum_{|\boldsymbol{x}-\boldsymbol{y}|=l} |f(\boldsymbol{x}) - f(\boldsymbol{y})|, \tag{4.5}$$

where l is the length of a pixel.

Now we are ready to present the likelihood measure. It is defined as

$$p(f) = Z^{-1} \exp(-\alpha TV(f)) \tag{4.6}$$

where Z and α are positive constants.

This can be clearly seen as a measure of the noise, because when the Total Variation increases the measure decreases and this means that the noise is going up, as we have seen in Example 1.3.1.

Finally, we can notice that unlike the previous metrics, it is a no reference metric because in its definition there is not the function which represents the not distorted signal.

Chapter 5

Introduction to nuclear medicine

5.1 Nuclear medicine preliminaries

The following definitions are taken from [8].

Before going on to explain the problem addressed, it is necessary to give an overview of the terms we are going to use several times and the context in which we find ourselves.

We are in the field of nuclear medicine, a specialized area of medical science that utilizes radioactive substances for diagnosis, treatment, and research.

This innovative branch of medicine combines principles of physics, chemistry, biology, and medicine to create advanced imaging and therapeutic techniques.

Indeed, by employing small amounts of radioactive materials known as radiopharmaceuticals, or radiotracer, which are substances that are part of the normal physiological processes in the body, with the difference that they are labeled with a radionuclide that can be detected by external means, nuclear medicine provides detailed insights into the structure and function of organs, tissues, and cells, enabling early detection and treatment of diseases.

Unlike conventional imaging methods that primarily show anatomical details, nuclear medicine reveals physiological and metabolic processes, offering a unique and comprehensive perspective on health and disease.

This field is at the forefront of medical diagnostics and therapy, continually evolving to improve patient care and outcomes.

In nuclear medicine, the most famous imaging methods used in clinical diagnosis are single-photon emission computed tomography (SPECT) and positron emission tomography (PET). Both are tomographic imaging methods that allow to study physiological processes by observing the distribution of the concentration of a radiotracer within the subject that is being studied.

Firstly, we have to fix the ideas of radiopharmaceuticals.

We have said that they are substances of the normal physiological processes in the body labeled with radionuclide.

A nuclide is a particular combination of nucleons (protons, neutrons). A radionuclide is a nuclide that naturally emits radiation, it is thanks to this radiation that bio-images will be produced.

The notation used to denote a given nuclide is ${}^{A}_{Z}X$, where A is the atomic mass number, or number of nucleons, Z is the atomic number, or number of protons, and X is the symbol of the chemical element. Examples of nuclides of interest in PET and SPECT are ${}^{18}_{9}F$ and ${}^{99}_{43}Tc$.

Nuclides can be grouped into families, which share some common attributes.

- Isotopes are nuclides that have the same Z or the same number of protons;
- Isobars are nuclides that have the same atomic mass number of A or the same number of nucleons;
- Isotones are nuclides that have the same number of neutrons;
- Isomers are different energy states of the same nuclide, isomers are when one or more of the nucleons are bound differently (in different energy states) between the two.

Each of these families plays a crucial role in nuclear medicine.

Stable nuclides are ones that do not naturally transform themselves and emit radiation in the process, while unstable nuclides are nuclides which naturally decay emitting radiation and trasforming themselves into a stable nuclide. We are interested in the latter.

We wonder when a nuclide is unstable.

A nuclide can be unstable due to several reasons related to its nuclear structure and the forces acting within the nucleus.

1. **Nuclear Binding Energy:**
    - In an atomic nucleus, protons and neutrons are bound together by the strong nuclear force. This binding is essential to hold the nucleus together against the repulsive electromagnetic force between positively charged protons.
    - The stability of a nucleus depends on the balance between the attractive nuclear force and the repulsive electromagnetic force. Nuclides that have an imbalance—either too many or too few neutrons relative to protons—may be unstable.
2. **Ratio of Neutrons to Protons (N/Z ratio):**

- Stable nuclei typically have a balanced ratio of neutrons to protons (N/Z ratio). This ratio varies depending on the element but generally increases with increasing atomic number.
- Nuclides with too many or too few neutrons relative to the number of protons (compared to stable isotopes of the same element) may undergo radioactive decay to achieve a more stable N/Z ratio.

5.1.1 Radioactive decays

We have said that bio-images are obtained through the collection of emitted radiation. Let us quickly deal with the six types of radioactive decays.

The first mode of decay is *alpha decay*. Alpha decay predominantly occurs when large nuclides decrease their mass through the emission of a ${}^{4}_{2}He$ nucleus (the alpha particle) to achieve greater binding energy per nucleon for the remaining nucleons.

The equation for alpha decay is given by

$$ {}^{A}_{Z}P \rightarrow {}^{A-4}_{Z-2}D + {}^{4}_{2}He $$

where P is the parent nuclide and D is the daughter.

The second mode of decay is β decay, which is also called negatron decay to distinguish it from positron decay, which is discussed next.

Beta decay is basically the turning of a neutron into a proton with the emission of the beta particle and an antineutrino

$$ n \rightarrow p + \beta^{-} + \nu $$

A beta particle is an electron, which is created at the moment of emission. In fact, outside of the nucleus neutrons are not stable and undergo beta decay to protons ${}^{1}_{1}H$ nuclei.

The equation for beta decay is

$$ {}^{A}_{Z}P \rightarrow {}^{A}_{Z+1}D + {}^{0}_{-1}e + \bar{\nu} $$

where ${}^{0}_{-1}e$ is the β^{-} particle or negatron, and $\bar{\nu}$ is the antineutrino.

The third mode of decay is positron decay, it is the conversion of a proton into a neutron by the emission of a positive electron (the positron) and a neutrino. Neutrons are larger in mass than protons, so this takes place inside of a nucleus, where this conversion leads to the daughter nuclide having its nucleons even more tightly bound (i.e., there is an increase in the binding energy). The equation for positron decay is

$$^{A}_{Z}P \rightarrow ^{A}_{Z-1}D + ^{0}_{+1}e + \nu$$

where $^{0}_{+1}e$ is the positron (β^{+}), and ν is the neutrino.

The fourth mode of decay is that of EC (electron capture). It occurs when one of the orbital electrons of the atom enters the nucleus and combines with a proton to convert it into a neutron. The equation for EC is

$$^{A}_{Z}P - ^{0}_{-1}e \rightarrow ^{A}_{Z-1}D + \nu$$

In this case, the neutrino carries away all of the energy of release so long as the daughter goes to the ground state. Some radionuclides that undergo EC can be used in SPECT imaging. This is because sometimes the decay is from an excited state of the daughter and the transition to the ground state leads to the emission of a photon, which can be used in imaging.

The fifth and sixth modes of decay are forms of IT (internal conversion). That is, they occur when an excited nuclide de-excites to the ground state or a state closer to the ground state. [8]

Example 5.1.1. Let us take the case of I123 as an example to fix our minds. I123 is an unstable isotope. The primary decay mode for I123 is electron capture, where a proton in the nucleus captures an orbiting electron and converts into a neutron, emitting a neutrino in the process:

$$^{123}I + e^{-} \rightarrow ^{123}Te + \nu.$$

Iodine-123 is used to label ioflupane, known as DaTSCAN, in nuclear medicine. DaTSCAN is the trade name for $[^{123}I]$ioflupane, a radiopharmaceutical used in single-photon emission computed tomography (SPECT) imaging.

The choice of ioflupane is because it binds with high affinity to the dopamine transporters (DaT) located on the presynaptic neurons in the striatum of the brain. Dopamine transporters are involved in the reuptake of dopamine from the synaptic cleft back into the presynaptic neuron, playing a critical role in dopaminergic neurotransmission.

Therefore, once administered, $[^{123}I]$ioflupane is taken up by the dopamine transporters. The gamma emissions from I-123 are detected using a SPECT camera, producing images that show the distribution and density of dopamine transporters in the brain.

By providing detailed images of dopamine transporter density, DaTSCAN can be used to monitor the progression of neurodegenerative diseases and the efficacy of therapeutic interventions.

In conclusion, if through BO we get the configuration of parameters that will lead to a better bio-image, we obtain a tool that can support the work of doctors.

5.2 Attenuation correction

The objective of SPECT is to provide a 3D view of radiopharmaceutical distribution in the body. However, since gamma rays are coming from the deep, they pass a certain quantity of material so they are attenuated and they cannot contribute to a precise reconstruction of the image.

Attenuation correction (AC) is therefore the method used to obtain a 3D image matrix that takes attenuation into account.

Attenuation correction has been addressed in nuclear medicine in several ways. One possibility is to correct for attenuation mathematically, after reconstructing the 3D image matrix. The end result of AC, whichever way it is accomplished, is to increase the counts/voxel values proportionally to the amount of attenuation experienced by that voxel.[7]

5.2.1 Attenuation and Scatter

The physics definition of *attenuation* is "a phenomenon in which the amount of detected radiation is reduced by scattering or absorption", so with attenuation we mean absorption or scattering.

In the case of narrow-beam geometry, it is known the attenuation equation:

$$I(\boldsymbol{x}) = I(0)e^{-\mu \boldsymbol{x}},$$

where I is the intensity, e.g., counts per unit time, measured either with no attenuating material present $I(0)$, or with $\boldsymbol{x}$ cm thickness of attenuating material present $I(\boldsymbol{x})$. The value for the linear attenuation coefficient μ depends on the energy of the photons in the beam, for the most part decreasing as energy increases. It varies also with the composition of the absorbing material.

However, this is not what happens in real life. Typically, gamma rays in the energy range used in nuclear medicine are more likely to undergo Compton-scattering interactions than photoelectric absorption in human tissue.

These interactions produce secondary gamma rays whose direction of travel is different than that of the original gamma ray. We would ideally be able to identify and exclude scattered gamma rays based on their loss of energy. But Compton scatter most often results in small scattering angles, such that many scattered gamma rays still have energies that are within the energy window. Therefore, the actual radiation intensity in the broadbeam situation is greater than the attenuation equation predicts.

One way to account for this discrepancy would be to include a multiplicative build-up factor in the attenuation calculation to account for the scattered gamma rays. Alternatively, we can determine an effective value

for the linear attenuation coefficient that takes broad-beam geometry into consideration: $I(\boldsymbol{x}) = I(0)e^{-\mu_{eff}\boldsymbol{x}}$.[7]

5.3 StarGuide

The description of the following parameters derives from [13].

In order to obtain a bio-image we have said that it is necessary to select some parameters and then the StarGuide provides to the respective bio-image. We proceed describing the parameters that we have used for the reconstructions.

The machine StarGuide supports noise reduction regularizations. The applied noise reduction methods are:

- **RDP (Relative Differences for Maximum)**: Penalizes relative differences between adjacent pixels, designed to avoid excessive uniformity at edges by attenuating penalties for large-scale differences.
- **MRP (Median Root Prior)**: Penalty is based on the deviation of a voxel from the local median in the adjacent area, assuming images are locally monotonic. This method reduces noise in locally monotonic regions and preserves edges.

The two parameters that deal with this are:

- **Beta parameter (for all regularization methods)**: Sets the penalty weight. Lower beta values preserve differences in the image better (sharper image), while higher beta values result in a more uniform image with less background noise.
- **Gamma parameter (for RDP regularization)**: Controls the shapes of prior assumption functions, essentially the scale of differences in the image considered as "signal" rather than "noise." Higher gamma values preserve differences in the image better (sharper image), while lower gamma values result in a more uniform image with less background noise.

The other two selectable parameters for reconstruction are iterations and subsets:

- **Iterations**: In biomedical reconstruction refer to the repeated application of an algorithm to refine the image estimate. During each iteration, the algorithm updates the image based on the current estimate and the measured data. This process continues until a stopping criterion is met, such as a specified number of iterations, convergence to a solution, or the achievement of a desired image quality.

- **Data Subsets:** The measured data (e.g., projections in PET or SPECT) are divided into smaller subsets. Each subset contains a portion of the total data. To accelerate the iterative reconstruction process, the data is often divided into subsets, and the algorithm processes these subsets sequentially within each iteration. This approach is known as Ordered Subsets Expectation Maximization (OSEM) or other similar techniques.

Chapter 6

Application of BO in SPECT reconstructions

6.1 Introduction to the optimization problem

The problem that is considered is to study the behavior of a new SPECT/CT machine, called StarGuide, sited in the Department of Nuclear Medicine in Padua. This machine produces a SPECT reconstruction which depends on a certain configuration of parameters, i.e., as the input parameters change, we obtain a different bio-image in the output. The aim is to find a reliable combination of input parameters in order to obtain the "best possible bio-image", where, by best possible bio-image, we mean the one that can help the doctor most in diagnosis.

To study this task, the first step is to create the bio-image that should work as a ground-truth, which is necessary to better analyze the optimization process in a controlled setting. Indeed, in concrete applications, such a ground-truth is not available. Then, we define a metric that compares this ground-truth with the bio-images produced by StarGuide. Finally, by using the mathematical algorithm discussed in Chapter 3, we try to look for the best configuration of StarGuide parameters that minimizes the metrics adopted. This approach will be developed step by step in detail in the next sections.

6.2 Resolution and results

For our discussion, a phantom filled with Iodine-123 (DatSCAN) was taken. The phantom has in its inside two volumes which have the aim to represent the striatum of the brain. These volumes are filled with a concentration of I123 which is approximately eight times the concentration of the background.

6.2.1 The role of the ground-truth in the experiment

The ground-truth is the information that is known to be real or true, provided by direct observation and measurement. We know exactly the density of the radiopharmaceutical which is inside the phantom, so the step that we have to do is to try to reproduce a bio-image that represents the ground-truth taking in consideration the fact that these values we know that are real will be modified by errors that occur during the measurement and attenuation of energy.

In order to construct the ideal bio-image, we have considered a CT of the phantom. From CT we obtain the contours. Now the idea is to fill the voxel of the CT with the values that we know (ground-truth) and then do the correction according to the attenuation.

The bio-image that we should obtain should ideally correspond to the image that a doctor should receive from the machine, that is, the image that the machine reconstructions should resemble.

By reconstruction that resembles the most to our artifice, we mean the combination of the parameters used for the reconstruction that minimize the score of a fixed distance from the artifice.

From now on, by ground-truth we will refer to the bio-image that should represent reality after Attenuation Correction.

Registration

One problem that could be faced during this process is the fact that the CT scan and SPECT scan are not aligned. This is a problem since some metrics that give an estimate of similarity between images need to have the voxels to be compared aligned. Therefore, it is important to use tools that allow the so called registration, a term used to indicate the alignment of images, even 3D images as in our case.

We used 3D Slicer for this step, software specialised in dealing with biomedical images.

Below we briefly outline how a quick alignment of biomedical volumes can be done.

3D Slicer is a software organised in modules. Each module has a particular function. One of the modules used for image registration is “Fiducial Registration”. When you switch to this module, on the left-hand side, you have the menus “Fixed landmarks” and “Moving landmarks”. By clicking on them, the option “Create new Point List as...” appears. This allows us to create two sequences of points to overlap in an orderly manner, in the sense that the first point of the sequence “Moving landmarks” will move on the first point of the sequence of “Fixed landmarks”, the second point of the sequence “Moving landmarks” will moved on the second point of the sequence of “Fixed landmarks” and so on. Then we use the “Save transform”

menu to save the transformation above and apply it in the "Data" module to the volume to be translated to obtain the desired overlap.

Segmentation

After the registration of the CT scan with the SPECT scan, the next step is to use the new CT to create the ideal SPECT bio-image. We use again 3D Slicer. It has a module which is capable of creating volumes and transforming them into biomedical images in the same format of that produced by StarGuide, i.e., DICOM.

Now we use the words *segment* and *segmentation*. By segment we mean a part of a volume and by segmentation a group of segments.

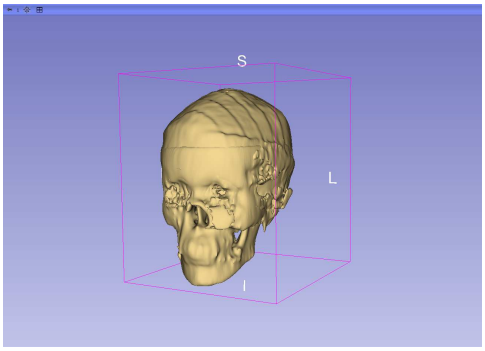
In this case, the module to accomplish this operation is "Segment editor". On the left of this module, press "Create new segmentation" to create the desired volume. "Source volume" allows to select the volume on which to base the biomedical image we are building. Now we can begin the construction of the volume. Click on "Add" to begin the creation of the new segmentation, so of a group of segments.

The interface presents useful tools for segmentation. The more important ones are:

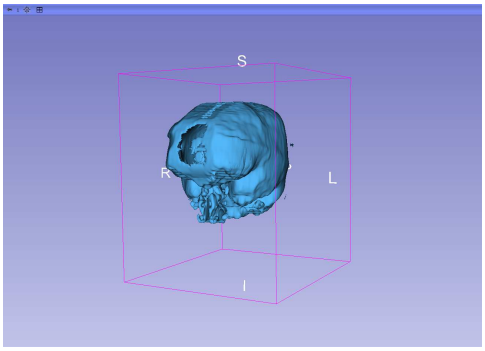
1. "Paint", which allows to highlight a certain part of a slice and by going slice by slice you can highlight the relevant parts to create the volume;
2. "Erase" to refine or delete parts;
3. "Threshold" which allows you to select all parts of the volume which have a value which belongs to a certain range of the grayscale;
4. "Scissors" to remove everything outside or inside the volume of selection.

"Threshold" is called semiautomatic because it allows to make a precise and fast selection while "Paint" must be used manually. Once that the segmentation is done, through the green arrow you can move on to export selecting the measure of a volume that the segmentation has to have, in this case we have to select a volume obtained from the StarGuide, and save the segmentation to a folder.

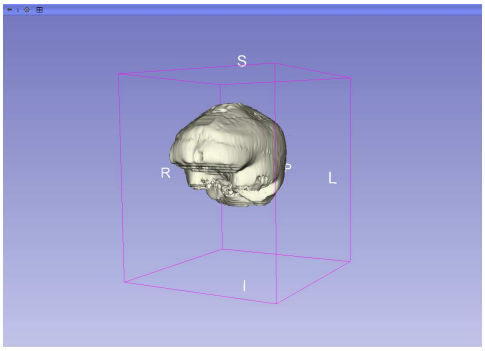
In Figure 6.1, we present an approximate segmentation of all the elements that compound our phantom.



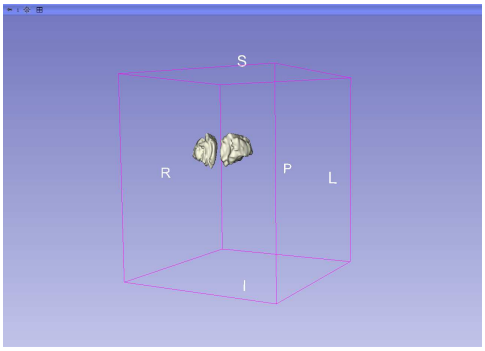
(a) Skull of the phantom.



(b) Cerebrospinal fluid of the phantom.



(c) Brain of the phantom.



(d) Striatum of the phantom.

Figure 6.1: Segmentation: parts which compound the phantom.

Once the segmentation is completed and exported, it appears in “Data”. We can save the still editable segmentation by doing “Save” in “Data” and selecting only the file with extension “.seg.nrrd”.

The segmentation is a file with the extension “.nrrd” in the selected folder. Now we have to transform it to DICOM, in order to get a volume of the same format of the bio-images produced by StarGuide. To do it the steps to follow are:

1. Open the folder in which there is the file “name-of-the-segmentation.nrrd” and drag it into 3D Slicer;
2. Click the “DCM” icon and select the file just uploaded;
3. Create a new study in “Data” and drag the volume under this study;
4. Right click and “Export to DICOM...” on the created study, select “Scalar volume” and now the volume into slices of the segmentation has been obtained.

We do the segmentation procedure individually for each part of the phantom that we want to consider, and then put it in a folder that will be read by MATLAB.

We have to export the segments separately because the DICOM files exported will be read by MATLAB as a 3D matrix with entries 1 and 0, the 1 correspond to the voxels in which the segment exists and the 0 to the voxels in which it does not exist. So before of putting them together, we have to change the values to the entries, otherwise the neighboring pixels of neighboring parts might be confused with other.

We select for each segment a number which has to identify that segment and through a script we multiply the 3D matrix with that scalar. Now, summing the 3D matrices, we can identify easily which voxels belong to a precise segment.

The next problem is that we are interested only in the interior part of the phantom because it is the only part that contains radioactivity, but the other segmentations of Figure 6.1 will not be discarded we will explain soon their role.

Furthermore, the values of the pixels that we have established for the interior have to be modified, because the values that we have associated to the voxels should correspond to the radioactivity. After this we have also to modify them another time because these values should take into account Attenuation Correction, here we have to consider the other segmentations because the gamma rays pass also the artificial skull and the artificial cerebrospinal fluid .[7]

Also Attenuation Correction has been implemented in MATLAB in a simple way, taking attenuation coefficients from [6]. In conclusion, after all this operations, we obtain the ground-truth $\boldsymbol{g}$.

6.2.2 Definition of the objective function

In Chapter 4 we have introduced some metrics which have the aim of comparing images. They can be obviously extended to 3D images considering the 3D images made of slices of 2D images.

In this subsection we define an objective function which is a convex combination of those metrics, and our aim is to find the bio-image that maximize it because its maximum will represent the "best" image. The function is defined in the following way

$$f(\boldsymbol{x}) = \alpha SSIM(\boldsymbol{x}, \boldsymbol{g}) - \beta MSE(\boldsymbol{x}, \boldsymbol{g}) + \gamma Z^{-1} \exp(-aTV(\boldsymbol{x})), \qquad (6.1)$$

where $\alpha, \beta, \gamma > 0$, $\alpha + \beta + \gamma = 1$, $\boldsymbol{g}$ is the ground-truth and $\boldsymbol{x}$ is the image that we are measuring.

Observation 6.2.1. Let us now analyze the expression of the function (6.1). It is convex combination of metrics. We know from the theory that

convex combination of convex or concave functions is convex or concave, respectively.

However, the functions that compound our combination are SSIM, of which we are unable to establish its concavity, MSE, which having the negative sign in front becomes a concave function, and the left branch of the exponential which is a convex function.

Unfortunately, we cannot establish a priori the presence of a maximum or a minimum for the concavity of the function.

However, Dominique Brunet and Edward R. Vrscay, in [2] provided to a reformulation of the SSIM in order to obtain convexity property.

After writing SSIM in the form:

$$SSIM(\boldsymbol{x}, \boldsymbol{y}) = S_1(\boldsymbol{x}, \boldsymbol{y}) S_2(\boldsymbol{x}, \boldsymbol{y}),$$

where

$$\begin{aligned} S_1(\boldsymbol{x}, \boldsymbol{y}) &= \frac{2\mu_{\boldsymbol{x}}\mu_{\boldsymbol{y}} + C_1}{\mu_{\boldsymbol{x}}^2 + \mu_{\boldsymbol{y}}^2 + C_1} \\ S_2(\boldsymbol{x}, \boldsymbol{y}) &= \frac{2\sigma_{\boldsymbol{xy}} + C_2}{\sigma_{\boldsymbol{x}}^2 + \sigma_{\boldsymbol{y}}^2 + C_2}, \end{aligned}$$

they defined the *normalized metrics*:[2]

$$\begin{aligned} d_1(\mu_{\boldsymbol{x}}, \mu_{\boldsymbol{y}}) : &= \sqrt{\frac{|\mu_{\boldsymbol{x}} - \mu_{\boldsymbol{y}}|^2}{\mu_{\boldsymbol{x}}^2 + \mu_{\boldsymbol{y}^2} + C_1}} \\ &= \sqrt{1 - S_1(\boldsymbol{x}, \boldsymbol{y})} \end{aligned} \tag{6.2}$$

$$\begin{aligned} d_2(\boldsymbol{x} - \mu_{\boldsymbol{x}}, \boldsymbol{y} - \mu_{\boldsymbol{y}}) : &= \sqrt{\frac{||(\boldsymbol{x} - \mu_{\boldsymbol{x}}) - (\boldsymbol{y} - \mu_{\boldsymbol{y}})||^2}{||\boldsymbol{x} - \mu_{\boldsymbol{x}}||^2 + ||\boldsymbol{y} - \mu_{\boldsymbol{y}}||^2 + (N-1)C_2}} \\ &= \sqrt{1 - S_2(\boldsymbol{x}, \boldsymbol{y})}. \end{aligned} \tag{6.3}$$

Combining (6.2) and (6.3) we obtain the vector

$$\boldsymbol{d}(\boldsymbol{x}, \boldsymbol{y}) = (d_1(\mu_{\boldsymbol{x}}, \mu_{\boldsymbol{y}}), d_2(\boldsymbol{x} - \mu_{\boldsymbol{x}}, \boldsymbol{y} - \mu_{\boldsymbol{x}})). \tag{6.4}$$

Their work proceed showing that

$$D(\cdot, \cdot) := ||\boldsymbol{d}(\cdot, \cdot)|| \tag{6.5}$$

is a metric, in particular a cone metric.

Finally, through Theorem 3.13 they show its convexity. The interest of demonstrating that (6.5) is a convex metric is because it can be seen as a low-order estimation of $\sqrt{1 - SSIM(\boldsymbol{x}, \boldsymbol{y})}$. Indeed, we have

$$\begin{aligned}\sqrt{1 - SSIM(\boldsymbol{x}, \boldsymbol{y})} &= \sqrt{1 - S_1 S_2} \\ &= \sqrt{d_1^2 + d_2^2 - d_1^2 d_2^2}.\end{aligned}$$

Furthermore, (6.5) can be seen as a likelihood measure because if either $\boldsymbol{x} = \boldsymbol{y}$ or $\boldsymbol{x} - \mu_{\boldsymbol{x}} = \boldsymbol{y} - \mu_{\boldsymbol{x}}$, we have respectively $S_1(\boldsymbol{x}, \boldsymbol{y}) = 1$ or $S_2(\boldsymbol{x}, \boldsymbol{y}) = 1$ and we have

$$||\boldsymbol{d}(\boldsymbol{x}, \boldsymbol{y})|| = \sqrt{1 - SSIM(\boldsymbol{x}, \boldsymbol{y})}.$$

In conclusion, if we want to obtain convexity property to take advantage of, we could substitute the $SSIM$, with the metric (6.5).

6.2.3 Application of BO and results

To apply BO, in particular Algorithm 1, we need a research set $\mathcal{X}$, i.e., a grid of evaluation such that each point corresponds to a configuration of parameters and so to a reconstruction.

However, one problem that we encountered was precisely the creation of this grid. This was due to the lack of automation of the software, since it is not possible to set a series of configurations of parameters and expect the respective bio-images.

Therefore, in order to create the grid, we had to set the desired parameter configuration each time and then create the corresponding bio-image to associate.

We did a grid of eighty-one reconstructions in about six hours.

Furthermore, another problem we faced was a non-consistency between measurements and theory.

We said we filled the phantom volumes, which correspond to the striatum of the brain, with eight times the concentration of the background. We have more than 1Bq per voxel. 1Bq is equivalent to one radioactive decay per second, and since the measurements were taken for fortyfive minutes, we should have about $8 \times 60 \times 45 = 21600$ decays in the pixels with the highest concentration, without taking into account the attenuation.

Below we report the distribution of the maximum number of hits detected from a voxel for each reconstruction.

On the horizontal axis we have the reconstructions, while, on the vertical axis the values detected.

Therefore, by the previous reasoning, we said that we should expect a value of about 10000 without considering attenuation. However, as we can see from Figure 6.2 this is not the case.

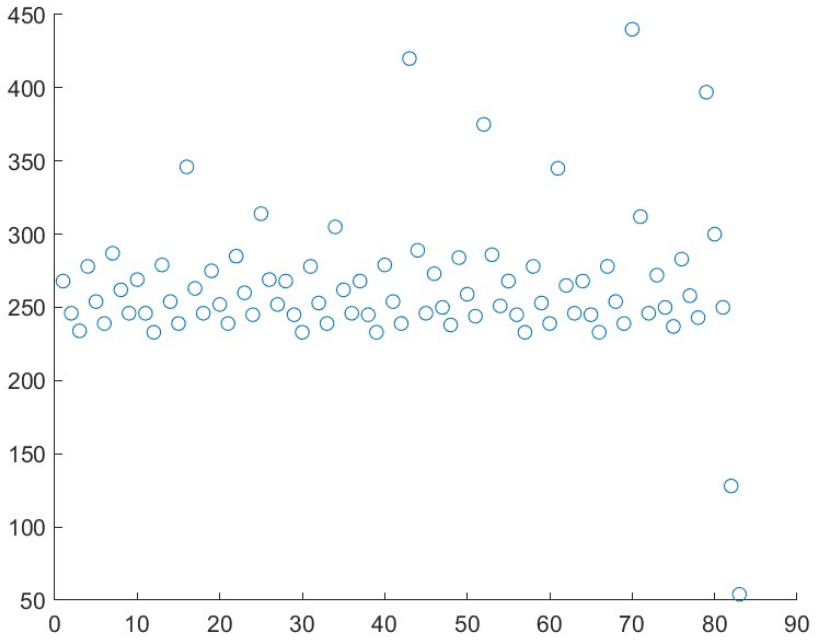


Figure 6.2: Distribution of the highest values obtained in each reconstruction.

From this simple observation we have learnt that the StarGuide makes a rescaling, unfortunately, technicians did not manage to give us an estimate of this change. However, this factor should multiply each voxel entry of the dicom files, consequently there should be a uniform rescale. As far as the negative MSE, SSIM and (4.6) are concerned, can be proved that the maximum is preserved after a uniform rescale.

We prove this only for the case of MSE.

Let $\boldsymbol{x}^k$ be the bio-image which minimizes the MSE with ground-truth $\boldsymbol{g}$, i.e.,

$$MSE(\boldsymbol{x}^k, \boldsymbol{g}) \leq MSE(\boldsymbol{x}^i, \boldsymbol{g}) \quad \forall i \in \{1, \dots, N\},$$

where N is the number of the available reconstructions.

Now we consider the the rescaled bio-image cx^i $\forall i$. We have to prove

$$MSE(c\boldsymbol{x}^k, \boldsymbol{g}) \leq MSE(c\boldsymbol{x}^i, \boldsymbol{g}) \quad \forall i \in \{1, \dots, N\}.$$

We substitute the expanded forms in the previous inequality:

$$\frac{1}{n}\sum_{j=1}^{n}(cx_j^k - g_j)^2 \leq \frac{1}{n}\sum_{j=1}^{n}(cx_j^i - g_j)^2,$$

where g_j and x_j^k are the numbers contained in the $j-th$ voxel of $\boldsymbol{g}$ and $\boldsymbol{x}^k$ respectively, and n the number of voxels. Rearranging the inequality by cancelling the common term $\sum_{j=1}^{n}(g_j)^2$, we obtain

$$-\frac{2c}{n}\sum_{j=1}^{n}x_j^k g_j+\frac{c^2}{n}\sum_{j=1}^{n}\left(x_j^k\right)^2\leq-\frac{2c}{n}\sum_{j=1}^{n}x_j^i g_j+\frac{c^2}{n}\sum_{j=1}^{n}\left(x_j^i\right)^2$$

After algebraic manipulation we get

$$-\frac{2}{c}\left(\frac{1}{n}\sum_{j=1}^{n}x_j^k g_j\right)+\left(\frac{1}{n}\sum_{j=1}^{n}\left(x_j^k\right)^2\right)\leq-\frac{2}{c}\left(\frac{1}{n}\sum_{j=1}^{n}x_j^i g_j\right)+\left(\frac{1}{n}\sum_{j=1}^{n}\left(x_j^i\right)^2\right)$$

Finally, multiplying the entire inequality by $\frac{c}{2}$

$$-\left(\frac{1}{n}\sum_{j=1}^{n}x_j^k g_j\right)+\frac{c}{2}\left(\frac{1}{n}\sum_{j=1}^{n}\left(x_j^k\right)^2\right)\leq\left(\frac{1}{n}\sum_{j=1}^{n}x_j^i g_j\right)+\frac{c}{2}\left(\frac{1}{n}\sum_{j=1}^{n}\left(x_j^i\right)^2\right)$$

which is true because by hypothesis we have

$$\frac{1}{n}\sum_{j=1}^{n}(x_j^k-g_j)^2\leq\frac{1}{n}\sum_{j=1}^{n}(x_j^i-g_j)^2\quad\forall i.$$

In conclusion, even if the output of the algorithm will give uncomfortable numbers, the convergence of the algorithm will still give a guarantee and this is due to the independence from a uniform rescaling factor.

The last clarification we need to make is that the dicom files are rescaled in the grayscale to produce the bio-image. This involves multiplying each reconstruction, and also our artifice, by the factor $\frac{255}{M}$ where M is the maximum number of hits of the considered reconstruction.

Finally, we are ready to apply (6.1) in order to compare reconstructions to the ground-truth.

Now we are ready to show and analyze the results. As a first experiment, we consider the optimization for $\alpha=1$, $\beta=0$, and $\gamma=0$. The results can be seen in Figure 6.3.

From Figure 6.3 we can see that during iterations the objective function does not vary on the visited nodes, this means that as far as SSIM is concerned, all reconstructions considered during the algorithm have almost constant evaluation when compared with ground-truth.

As second experiment, we consider the setting $\alpha=0$, $\beta=1$, and $\gamma=0$. The output of our algorithm is presented in Figure 6.4.

Unlike the previous experiment, we observe in Figure 6.4, that this one exhibits several oscillations. This indicates that the error derived from our

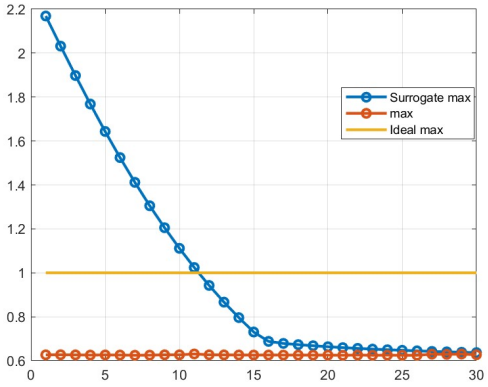


Figure 6.3: Comparison between surrogate model and objective function with setting of parameter $\alpha = 1$, $\beta = 0$, and $\gamma = 0$.

ground-truth artifice does not closely match, by MSE, the reconstructions obtained from the StarGuide and we can investigate for the most similar bio-image.

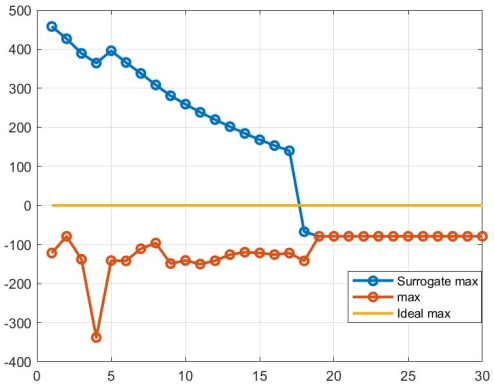
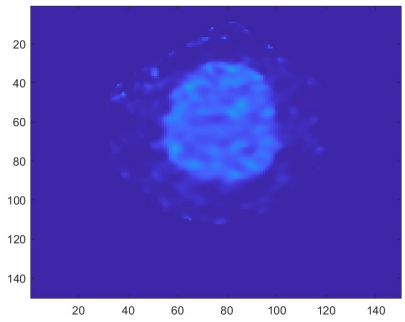
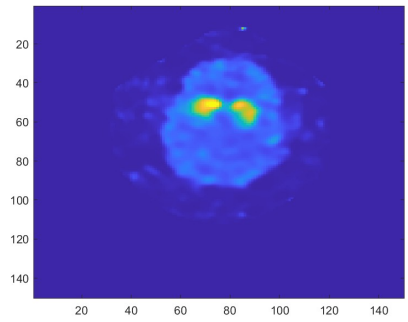


Figure 6.4: Comparison between surrogate model and objective function with setting of parameter $\alpha = 0$, $\beta = 1$, and $\gamma = 0$.

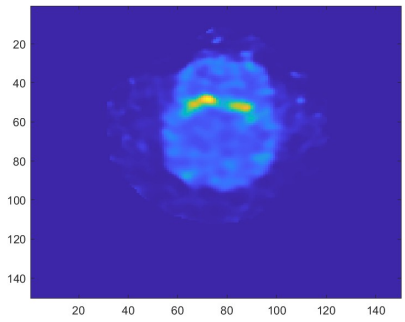
Also in this case we have the convergence between the surrogate model and the objective function among the visited nodes. Finally, we show in Figure 6.5 the reconstruction that is the outcome of the algorithm with $\alpha = 0$, $\beta = 1$, and $\gamma = 0$.



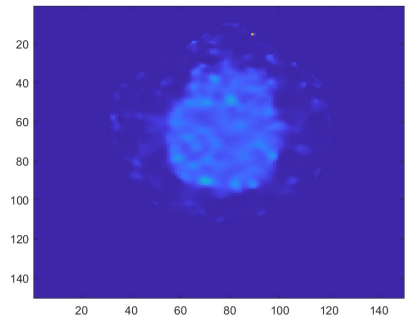
(a) 35th slice of the optimal reconstruction.



(b) 40th slice of the optimal reconstruction.



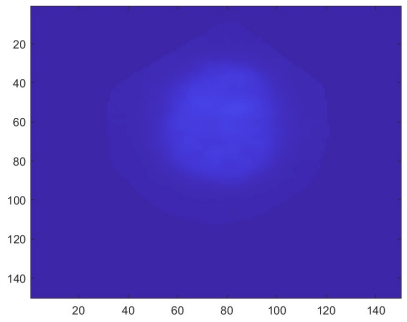
(c) 45th slice of the optimal reconstruction.



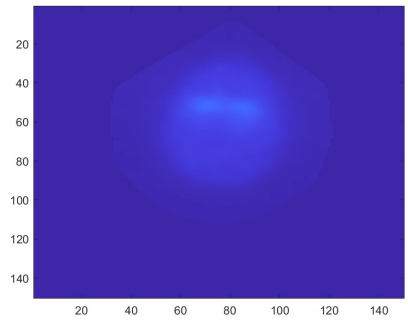
(d) 50th slice of the optimal reconstruction.

Figure 6.5: Optimal reconstruction obtained for $\alpha = 0$, $\beta = 1$, and $\gamma = 0$.

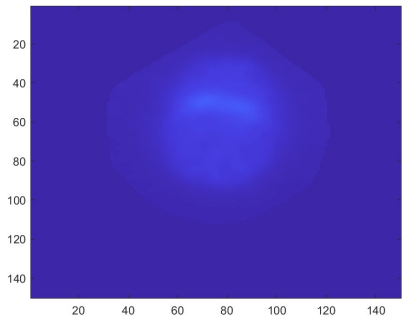
The meaningfullness of our optimization process can be supported by comparing Figure 6.5 to Figure 6.6, where the reconstruction that the algorithm visits at the fifth iteration, in which the objective function assumes its minimum, is displayed.



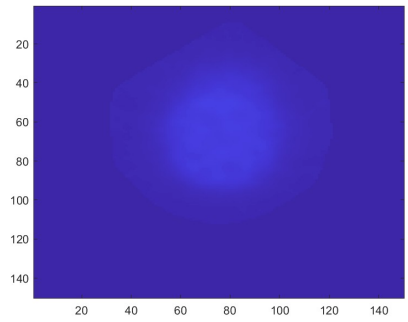
(a) 35th slice of the minimum of the visited nodes.



(b) 40th slice of the minimum of the visited nodes.



(c) 45th slice of the minimum of the visited nodes.



(d) 50th slice of the minimum of the visited nodes.

Figure 6.6: Minimum of the reconstruction obtained for $\alpha = 0$, $\beta = 1$, and $\gamma = 0$.

Then, we display the results for $\alpha = 0$, $\beta = 0$, and $\gamma = 1$. As we can see in Figure 6.7, we have the convergence of the algorithm to the real function starting from the 36th iteration. In this case, we remark that we are not comparing the reconstruction to the ground-truth.

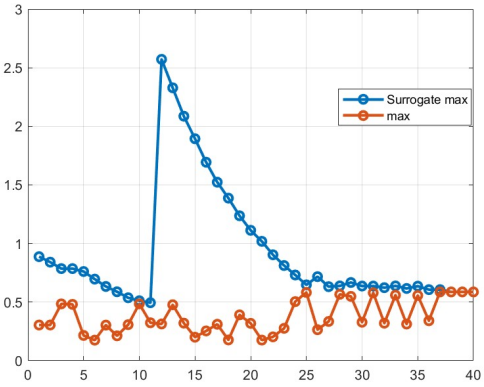
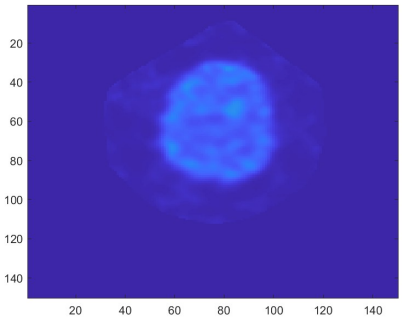
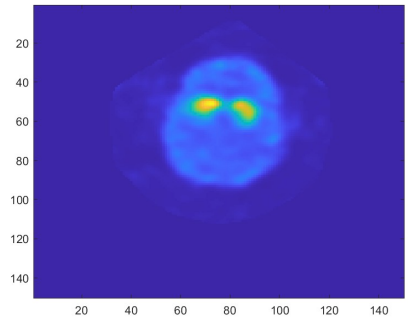


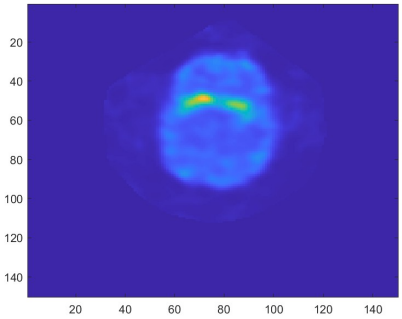
Figure 6.7: Comparison between surrogate model and objective function with setting of parameter $\alpha = 0$, $\beta = 0$, and $\gamma = 1$.



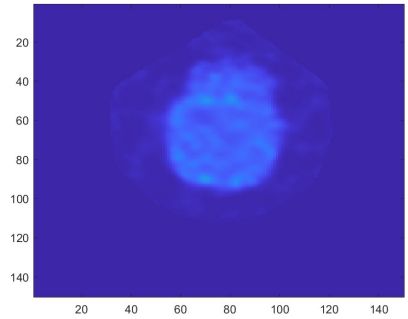
(a) 35th slice of the optimal reconstruction.



(b) 40th slice of the optimal reconstruction.



(c) 45th slice of the optimal reconstruction.



(d) 50th slice of the optimal reconstruction.

Figure 6.8: Optimal reconstruction obtained for $\alpha = 0$, $\beta = 0$, and $\gamma = 1$.

It might be difficult to discern the difference between the reconstruction shown in Figure 6.5 and the one presented here. However, upon closer inspection, it can be noticed that the voxels around the brain and the brain itself exhibit a more uniform color.

This is consistent with what one would expect from the measure defined on total variation. It is true that, as it is shown in the example 1.3.1, the decrease of Total Variation produces a "cleaner" image, but this could lead to a loss of contrast between image features, and this is precisely what happens in Figure 6.8.

A good idea may be using a combination of MSE and the (4.6) measure. Doing this, we try to preserve the contrast between features, thanks to MSE with ground-truth, and the sharpness of the image, thanks to (4.6).

After some manual parameter tuning, we arrived at the following outcome. In Figure 6.9 we present the comparison of surrogate model and objective function between the nodes viewed during the iterations. Again we have the convergence of the behavior of the surrogate model to the objective function.

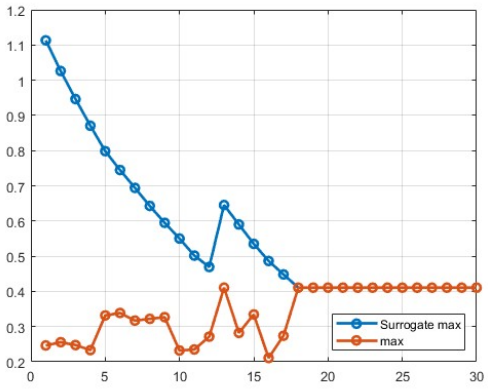
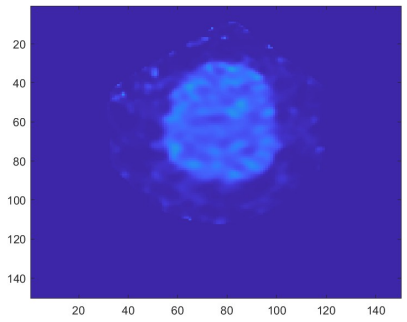
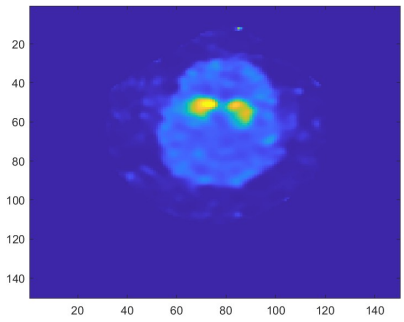


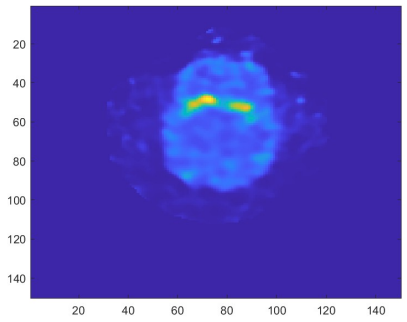
Figure 6.9: Comparison between surrogate model and objective function with setting of parameter $\alpha = 0.400$, $\beta = 0.001$, and $\gamma = 0.599$.



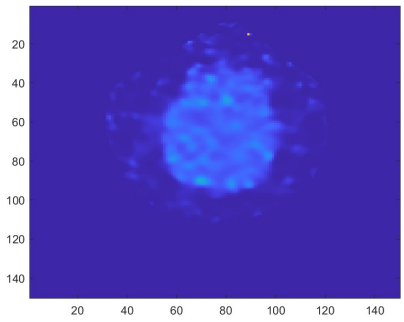
(a) 35th slice of the optimal reconstruction.



(b) 40th slice of the optimal reconstruction.



(c) 45th slice of the optimal reconstruction.



(d) 50th slice of the optimal reconstruction.

Figure 6.10: Optimal reconstruction obtained for $\alpha = 0.400$, $\beta = 0.001$, and $\gamma = 0.599$.

Conclusions

In this thesis, we effectively applied Bayesian optimization, grounded in Reproducing Kernel Hilbert Spaces, to optimize the parameters' configuration of the StarGuide SPECT/CT imaging machine.

Starting with a thorough review of the mathematical preliminaries necessary to understand BO and RKHS, we then detailed the BO algorithm and provided some examples to show its effectiveness in dealing with possibly non-concave functions.

The practical application of BO to the StarGuide machine, as detailed in Chapter 6, illustrated the specific challenges encountered and the solutions developed through BO. The results obtained demonstrate improvements in image quality, validating the hypothesis that BO can be a suitable tool in this context.

In conclusion, this work paves the way for a more accurate study of this optimization approach in this field of application, which may lead to better diagnoses, better patient outcomes, therefore significant implications in clinical practice in the Nuclear Medicine field. Moreover, this thesis is an example that reaffirms the value of interdisciplinary approaches in advancing healthcare technologies.

Bibliography

[1] Arthur Leclaire, Lionel Moisan, *No-Reference Image Quality Assessment and Blind Deblurring with Sharpness Metrics Exploiting Fourier Phase Information*, Springer, 2014.

[2] Dominique Brunet, Edward R. Vrscay, Member and Zhou Wang, *On the Mathematical Properties of the Structural Similarity Index*, ,2012.

[3] Giuseppe De Marco, *Analisi 2*, Edizioni Libreria Progetto, 2018.

[4] Gregory E. Fasshauer, *Meshfree Approximation Methods with MATLAB*, World Scientific Publishing Co.Pte. Ltd., 2007.

[5] Gwendoline Blanchet, Lionel Moisan, Bernard Rougé. *Measuring the global phase coherence of an image*. 15th IEEE International Conference on Image Processing (ICIP), 2008, San Diego, United States. pp.1176 - 1179, ff10.1109/ICIP.2008.4711970ff. ffhal-00212337 .

[6] J. H. Hubbell and S. M. Seltzer, *X-Ray Mass Attenuation Coefficients*, U.S. Secretary of Commerce on behalf of the United States of America, 2004.

[7] Jennifer Prekeges, *Nuclear medicine instrumentation*, Jones & Bartlett Publishers, 2013.

[8] Magnus Dahlbom, *Physics of PET and SPECT Imaging*, CRC Press, 2017.

[9] Niranjan Srinivas, Andreas Krause, Sham M Kakade, and Matthias Seeger, *Gaussian process optimization in the bandit setting: No regret and experimental design*, 2010.

[10] Oppenheim, A.V., Lim, J.S.: *The importance of phase in signals*, Proc. IEEE 69, 529–541, 1981.

[11] R. S. Ellis., *Measuring the global phase coherence of an image* , Springer-Verlag New York, 2006.

[12] Roger L. Easton, Jr., *Fourier Methods in Imaging*, Wiley, 2010.

[13] *StarGuide manuale dell'operatore*, General Electric Company, 2021

[14] Wendland, H., *Scattered Data Approximation*, Cambridge University Press (Cambridge), 2005.

[15] Yueming Lyu, Yuan Yuan , Ivor W. Tsang , *Efficient Batch Black-box Optimization with Deterministic Regret Bounds* , v3, publisher, 2020.

[16] Zhou Wang, Member, IEEE, Alan C. Bovik, Fellow, IEEE Hamid R. Sheikh, Student Member, IEEE, and Eero P. Simoncelli, Senior Member, IEEE, *Image Quality Assessment: From Error Visibility to Structural Similarity*, VOL.13, 2004.

Ringraziamenti

Non so se sono bravo con le parole ma ci provo.

Desidero ringraziare il mio relatore Francesco Marchetti, che è riuscito ad esaudire un mio sogno nel cassetto, applicare la matematica alla medicina, senza di lui questo lavoro non esisterebbe.

Ringrazio Alessandra, Diego e Luca per il supporto che mi hanno dato e perché che mi hanno permesso di entrare in contatto con un mondo diverso da quello a cui sono abituato.

Grazie Gianluca, grazie Giorgio, siete stati i primi ad avermi accettato e con cui ho davvero stretto un legame, probabilmente non vi merito ma cercherò di rimediare.

Grazie Francesco, forse non lo sai ma sei probabilmente una delle persone che stimo di più e forse una di quelle in cui più mi rivedo. Se sono qui, pensa un pò, è anche merito tuo.

Grazie Giulia, in questi anni di magistrale ci sei sempre stata. Hai sempre dato il massimo per tutti senza chiedere mai nulla in cambio. Questo è ciò che ti rende unica, non le altre sciocchezze dici.

Grazie Carlo, mi hai alleggerito il peso dello studio ogni sera insieme ai ragazzi, non so se ce l'avrei fatta senza di te.

Grazie Antonia, ci sono poche cose di cui sono certo, ma di te, lo sono. Nonostante il poco tempo mi hai già fatto crecere tanto, spero tu possa continuare a farlo. Sei ciò di cui ora ho bisogno.

Grazie alla mia famiglia, zia Ada, nonna Raffaella, nonno Sergio, zio Gianpietro, zia Sara e Claudia mi avete cresciuto e visto crescere, mi avete sempre dato il meglio di voi.

Grazie Nicola, per me sei più di un fratello. Sei la persona di cui probabilmente più mi fido.

Infine grazie alle persone a cui devo tutto, grazie ai miei genitori. Magari non mi avete insegnato la matematica, ma mi avete insegnato la costanza, il sacrificio e l'umiltà. In questi valori mi rivedo, non avrei potuto chiedere dei genitori migliori. Un grazie non sarà mai abbastanza. Spero, almeno un pò, di avervi resi orgogliosi di me.