Bachelor's Thesis

# Bayesian Optimisation - a Decision-Theoretic Perspective

Department of Statistics
Ludwig-Maximilians-Universität München

**Duc-Anh Nguyen**

Munich, th, 2024

Submitted in partial fulfillment of the requirements for the degree of B. Sc.
Supervised by Julian Rodemann

## Abstract

With such an ambitious title, this thesis wants to explain the art of Bayesian Optimisation, from the perspective of the one who studies Decision Theory. It will mainly focus on showing how one can apply some of the most basic, common known criteria from decision theory to some (also quite) basic optimising problems of Bayesian Optimisation (single point proposal, not noisy, not multi-objective, not mixed-space optimisation). More exactly, we want suggest a number of new acquisition functions inspired by a few desicion criteria such as: the maximax criterion, the maximin criterion, the minimax Regret cirterion, the pessimism—optimism index criterion of Hurwicz and the Hodges-Lehmann criterion. These new acquistions function will then be benchmarked and compared with the common ones, e.g.: Expected Improvement (EI) and Lower Confidence Bound (LCB).

Key words:

# Acknowledgements

# Contents

# 1 Introduction

This thesis focuses on showing the link between Decision Theory and Bayesian Optimisation (BO), in other words, how one could use Decision Theory to design effective BO strategies for optimising mathematical (both theoretic and practical) problems.

A good way to open this Introduction chapter is to briefly explain the idea of BO and the basic characteristics of Decision Theory. BO is widely used as an effective way to find solutions for the so-called design problems. One example of design problems would be in applied science, where engineers design different components of machines for more effective execution; or where scientists design experiments to learn more about some particular phenomena. Another example would be software design problems: how one develops software so that it would run smoothly in real-time with the least computational power. All of these design problems often involve a lot of options, which are often high dimensional, and usually also have interactions with each other, which would make it challenging to find the best solution.

The applications of BO are varied: Brochu et al. (2010) presents extensions of BO with experiments with hierarchical reinforcement learning; Snoek et al. (2012) and Thornton et al. (2013) shows different ways one could apply BO into tuning the hyperparameters in the field of machine learning. In robotics: Lizotte et al. (2007) discusses about Automatic Gait Optimisation using Gaussian Process Regression while Martinez-Cantin et al. (2007) explains the Active Policy Learning for Robot Planning.

We will discuss the most important components of a Bayesian Optimisation, then how to choose these components based on some criterion from decision theory.
TODO: write some more details here
The paper is structured as follows: in Chapter 2 we discuss all the crucial components making a typical **Bayesian Optimisation** strategy. Chapter 3 is where the foundation of **Decision Theory** will be explained in details (to some extends). Based on that, we will come up with some new methods (infill functions) combining knowledge in both fields in Chapter 4. In Chapter 5 we put these methods into use by benchmarking their usages in an experiment. Finally, Chapter 6 is preserved for concluding remarks.

# 2 Fundamental principles of Sequential Model-Based Optimisation

## 2.1 Bayesian Optimisation

Bayesian Optimisation (BO) has quite a long history, which dates back to the early 1930s, when Thompson's work pointed out the exploration-exploitation trade-off, a crucial idea when working with BO (see Thompson (1933), Shahriari et al. (2016)). The term BO was first used in the 1970s by Mockus et al. (2014).

So what exactly should one know about the method of BO to understand this thesis? Mathematically, BO tries to find a global maximiser (or minimiser) of an unknown objective function. In this thesis, we will focus on finding the **global (!) minimum** of a function, hence the general mathematical expression is stated:

$$x^* = \arg\min_{x \in \mathcal{X}} f(x)$$

with

$$\mathcal{X} = \mathcal{X}_1 \times \mathcal{X}_2 \times \mathcal{X}_3 \times ...\mathcal{X}_d$$

a **d**-dimensional design space of interest. If one wants to maximise a function g instead, the problem can now be regarded as the minimisation of the transformed function

$$f(x) = -g(x)$$

.

Here in the expression, one should understand that $\mathcal{X}$ could take values of type numerical, as well as of type categorical, but in this thesis, $\mathcal{X}_i$ is defined as a compact subset of real numbers set **R**. There are also a couple of properties and assumptions that should be mentioned here (also see Frazier (2018)):

- The target function f is assumed to be black-box, which has no closed form (problematic for finding extremum), hence no first- or second-order derivatives - gradient descent, Newtons method, or quasi Newton methods are not plausible here.
- However, it could be evaluated at any query point x in the domain $\mathcal{X}$, although evaluations are normally cost-sensitive - the number of available evaluations should be limited (typically a few hundreds of people), since each evaluation normally takes a lot of time/money.
- The dimension d should not be too large (smaller than 20) for BO to work properly.
- Another assumption is, that the objective function is Lipschitz-continuous. That is, that there exists some constant C, such that for all $x_1$, $x_2$ ∈X: $||f(x_1) - f(x_2)|| \leq C||x_1 - x_2||$ with C maybe unknown.

One more thing to mention here is, that in many scenarios, the evaluations' output are stochastic (corrupted with noises), but in this thesis we will only consider the cases without noises.

Since it is expensive to evaluate these functions, the goal is to find the optimum with the least amount of points to be evaluated. A **sequential model-based** approach to solving this stated problem is **BO** (see Hutter et al. (2011)). But first, what is a **sequential approach**?

Generally, a sequential search algorithm runs N queries, at each query, e.g.: at iteration n it selects a $x_{n+1}$, evaluate function f to get $y_{n+1}$, and in the very end it selects the best possible evaluation of the optimiser. According to Betrò (1991) and Brochu et al. (2010), the evaluation of an objective function (even in a noise-free domain), with Lipschitz continuity C on a d-dimensional unit hypercube, for the best observation $f(x^+)$ $\geq f(x^*) - \epsilon$ requires $(C/2\epsilon)^d$ samples, which is considered expensive.

Then it comes to the question of: what is **sequential model-based optimisation** (SMBO)? As the name suggests, SMBO methods would fit a regression model (response surface model) that later will be use for optimisation. (see Jones (2001)). A SMBO strategy is a combination of many components: designing each of these components can be done separately and then put together to set up a strategy that works best for a specific optimisation problem. This is considered as an **advantage** of using the SMBO.

More exactly, in case of BO, we will have two main components: a **probalistic surrogate model** and **an acquisition function**. We apply the **Bayesian Theorem** to create SMBO strategy: over the possible candidate objective functions we prescribe a prior belief (here the initial surrogate model) and then sequentially refine this model as data are observed via Bayesian posterior updating. The Bayesian posterior represents our updated beliefs in the candidate objective function we are optimising. Details will be covered in chapter 2.2.

Alongside this probabilistic model, we can sequentially induce **acquisition functions** which leverage the uncertainty in the posterior to manage a trade-off between exploration and exploitation. More on this will be discussed in chapter 2.3.

## 2.2 Surrogate Model

Surrogate models, as the name might suggest, are used as a cheaper alternative to evaluate the samples, since the target black box functions are usually considered cost-sensitive. Using the Bayesian principle, we assume that the target functions follow a certain distribution, from which we fit a surrogate model with the available initial sample points. We then update this model every time we get new sample points (refining the model).

For the sake of completeness, it should be mentioned that there are also parametric surrogate models (such as the Beta-Bernoulli bandit model using Thompson sampling or generalised linear models, see Shahriari et al. (2016) section II). However, the **focus of this thesis** lies on the **non-parametric** ones, in particular Gaussian Process Models. **Random Forests** would also be a common choice, especially when sample points contain a mixture of categorical and numerical features; we would not discuss this topic further - for more information, one could read Guo et al. (2023) and Ma et al. (2023).

### 2.2.1 Gaussian Process Regression

Gaussian Process Regression (GPR) is the most popular choice for the surrogate model for a Bayesian Optimisation strategy, because of one characteristic: these models do not only return a mean estimate, but also a variance. They describe a **probability distribution over all possible functions** that fit a certain set of points: for that we can compute the **means** as the **maximum likelihood estimate** of the function, and the **variances** as an indicator of **prediction confidence**.

Brochu et al. (2010) defines: A GP is an extension of the multivariate Gaussian distribution (or multivariate normal (MVN) distribution) to an infinite dimension stochastic process for which any infinite combination of dimensions will be a Gaussian distribution. Just as a Gaussian distribution is a distribution over a random variable, completely specified by its mean function, m and covariance function, k:

$$f(x) \sim \mathcal{GP}(\mu(x), k(x, x'))$$

with $\mu(x) = \mathrm{E}[\mathrm{f(x)}]$ a covariance/kernel function $k(x, x') = \mathbb{E}\left[(f(x) - \mu(x))(f(x') - \mu(x'))\right]$ and a common assumption: prior mean is a zero function: $\mu(x) = 0$; another assumption for the prior mean could be found here Martinez-Cantin et al. (2009). We construct the covariance matrix by evaluating a covariance function or kernel 0 at each pair of points $x_i$, $x_j$. The kernel is chosen so that points $x_i$, $x_j$ that are **closer** in the input space have a **larger** positive correlation, encoding the belief that they should have more similar function values than points that are far apart. The kernel should also have the property that the resulting covariance matrix is positive semi-definite (p.s.d.), regardless of the collection of points chosen. For this, there are a couple of common choices, one of which would be the squared exponential function:

$$k(x_i, x_j) = exp(-\frac{1}{2}||x_i - x_j||^2)$$

This is one very popular and intuitive kernel, as this function converges to 1 when the two values of $x_i$ and $x_j$ get closer together and to zero otherwise. We need this as a Bayesian optimisation method will converge to the optimum if the distance from one point to the nearest observation is zero, according to Mockus (1994) (also understood as conditional variance converges to zero). Other mentioned condition for the convergence would be, that the acquisition function is continuous and approximately minimizes the risk (the expected deviation from the global minimum at a fixed point x). Also, more sophisticated kernels will be discussed later on.

So where do we start? We start constructing our surrogate function by sampling from a prior, as any Bayesian method depends on a prior distribution, by definition. We choose $\{x_{1:t}\}$ and evaluate the mean at each point $x_i$ and covariance function at each pair $x_i$ and $x_j$, so in the end we sampled the pairs $\{x_{1:t}, f_{1:t}\}$. With the zero function as the prior mean, we assume that the values of the surrogate function are drawn according to a MVN distribution $N(0.\mathbf{K})$, with $\mathbf{K}$ the kernel/ covariance matrix:

$$\mathbf{K} = \begin{bmatrix} k(\mathbf{x}_1, \mathbf{x}_1) & \cdots & k(\mathbf{x}_1, \mathbf{x}_t) \\ \vdots & \ddots & \vdots \\ k(\mathbf{x}_t, \mathbf{x}_1) & \cdots & k(\mathbf{x}_t, \mathbf{x}_t) \end{bmatrix}$$

Now that we got the initial prior surrogate function, we would wish to refine this function every time we sample a new position from the sample space of our target model. Suppose that we sampled this new position $\{x_{t+1}, f_{t+1}\}$, then we would get this distribution that $f_{1:t}$ and $f_{t+1}$ are jointly Gaussian, applying the properties of Gaussian Processes:

$$\begin{bmatrix} f_{1:t} \\ f_{t+1} \end{bmatrix} \sim \mathcal{N}\left(0, \begin{bmatrix} \mathbf{K} & \mathbf{k} \\ \mathbf{k}^T & k(\mathbf{x}_{t+1}, \mathbf{x}_{t+1}) \end{bmatrix}\right),$$

with k = $\begin{bmatrix} k(\mathbf{x}_{t+1}, \mathbf{x}_1) & k(\mathbf{x}_{t+1}, \mathbf{x}_2) & \cdots & k(\mathbf{x}_{t+1}, \mathbf{x}_t) \end{bmatrix}$

Here we want to calculate the posterior distribution function (an updated surrogate model). We may then compute the conditional distribution of f(x) given these observations using Bayesian rule (more details see Rasmussen and Williams (2005)):

$$P(f_{t+1}|\mathcal{D}_{1:t}, \mathbf{x}_{t+1}) = \mathcal{N}\left(\mu_t(\mathbf{x}_{t+1}), \sigma_t^2(\mathbf{x}_{t+1})\right)$$

where

$$\mu_t(\mathbf{x}_{t+1}) = \mathbf{k}^T\mathbf{K}^{-1}\mathbf{f}_{1:t}$$
$$\sigma_t^2(\mathbf{x}_{t+1}) = k(\mathbf{x}_{t+1}, \mathbf{x}_{t+1}) - \mathbf{k}^T\mathbf{K}^{-1}\mathbf{k}.$$

are the sufficient statistics of the predictive posterior distribution: $P(f_{t+1}|\mathcal{D}_{1:t}, \mathbf{x}_{t+1})$. Gaussian Process Regression (GPR) with the metric of uncertainty is very suitable for SMBO (more on this in chapter 2.3). The predictions of the model are also considered easy to compute, as the number of query points is realtive small (in the sequential decision making setting).

### 2.2.2 More on kernels/ covariance functions

What does a kernel tell us? Basically, a kernel shows the structure of the response functions we can fit, especially the smoothness of the drawn samples. While the prior mean provides a possible offset, the kernel function controls the smoothness and amplitude of samples from the GP. There is one more relevant concept: stationary kernels, which do not change its properties in case of shifting. In this thesis we would focus on Matérn kernels, are a very flexible class of stationary kernels. Here are some of the most common Matérn kernels:

$$k_{\text{Mátérn1}}(\mathbf{x}, \mathbf{x}') = \theta_0^2 \exp(-r)$$
$$k_{\text{Mátérn3}}(\mathbf{x}, \mathbf{x}') = \theta_0^2 \exp\left(-\sqrt{3}\, r\right)\left(1 + \sqrt{3}\, r\right)$$
$$k_{\text{Mátérn5}}(\mathbf{x}, \mathbf{x}') = \theta_0^2 \exp\left(-\sqrt{5}\, r\right)\left(1 + \sqrt{5}\, r + \frac{5}{3} r^2\right)$$
$$k_{\text{SQ-EXP}}(\mathbf{x}, \mathbf{x}') = \theta_0^2 \exp\left(-\frac{1}{2} r^2\right),$$

where $r^2 = (\mathbf{x} - \mathbf{x}')^T diag(\theta^{-2})(\mathbf{x} - \mathbf{x}')$ and is a diagonal matrix of d squared length scales $\theta_i^2$. This family of covariance functions are therefore parameterized by an amplitude and d length scale hyperparameters, jointly denoted $\theta$. Some more insights of each kernels could be found in this tutorial Brochu et al. (2010). In Figure 1 we see a visualization of the kernel profiles and samples from the corresponding priors and posteriors.

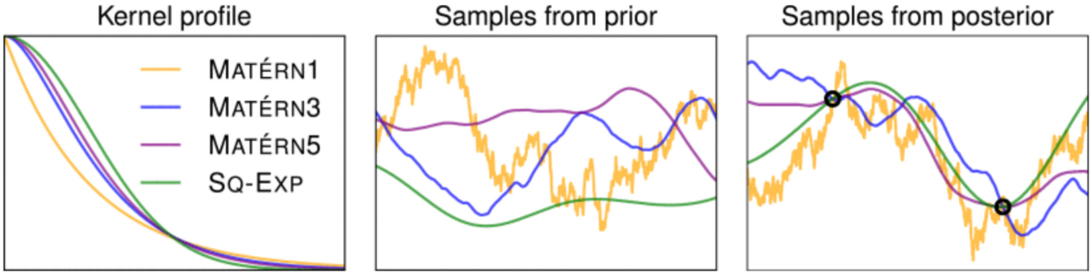

Figure 1: Left: common kernel profiles visualised, with the horizontal axis represents the distance $r > 0$. Middle: GPR prior functions with different kernels; with the squared exponential kernel the output function is the smoothest of all four, and with the Matérn the roughest. Right: Updated posterior functions given two new points (black circles). Source: Brochu et al. (2010)

## 2.3 Acquisition function

We have shown the statistical model representing our belief about the black-box function f at iteration n. What we need now is a policy for selecting the next query points. As throughout the thesis, we focus on how to find out the optimum with the least time we have to sample (since it is cost sensitive), it is salient that we choose our acquisition function wisely. Technically, the decision coming from infill functions represents an automatic trade-off between exploration (where the objective function is very uncertain about its predictions (with high posterior variance/ standard deviation)) and exploitation (with the smallest posterior mean, in case of minimisation). The acquisition function is typically designed so that a high acquisition corresponds to potentially the best optimal position in the sample space (in our case: lowest value of acquisition function means optimal location to sample from; whether it is the location with the optimum (lowest value in our case), or the location with great uncertainty (where it might reveal the most information about our black-box target function)).

Let's start with the most naive infill function, the **pure mean** optimisation function. It could be expressed as: $\alpha_{mean}(x; \mathcal{D}_n) = \mu_n(x)$ and our goal is to find $x^* = \arg\min_{x \in \mathcal{D}} \alpha_{mean}(x|\mathcal{D})$. It is naive though straightforward, as we will choose the location where the posterior mean has the lowest value, without considering the posterior variance / standard deviation. It should be considered too naive/ not optimal, as we will very likely only find **local optima** instead of **global optimum**. Figure 2 demonstrates exactly the problem.

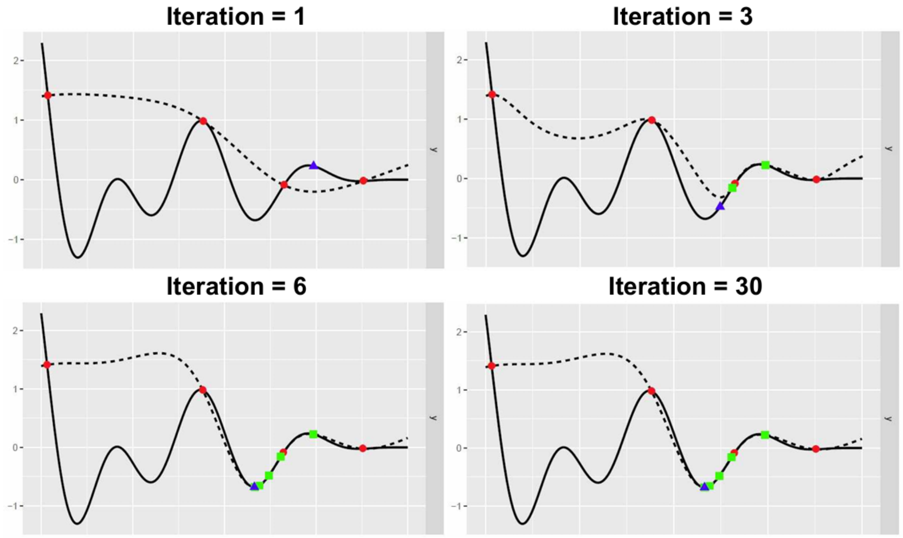

Figure 2: Here the red points are the initials sampled points for the initial surrogate function. The triangular blue point shows the location where to evaluate next. The green points are the newly evaluated location. The dotted line is our surrogate function and the black line is the black-box objective function. The problem of **local optimum** is visualised: although the location of our **global optimum** is on the left side, we are stuck on the right side of the plot for the whole 30 evaluations as the posterior mean always indicate that is where the optimum lies. Source: Mayer (2017)

With that example we want to emphasise the need of considering the posterior acquisition function as well. We will now discuss some of the most common infill functions in BO while categorising them according to their main goal.

### 2.3.1 Improvement-based policies

These infill functions favour the locations where the value of objective function are likely to improve upon the incumbent optimum. There are two acquisition functions worth mentioning, which are Probability of Improvement (PI) and Expected Improvement (EI).

The early work of Kushner (1964) on a ”new” method of locating the Maximum Point of an Arbitrary Multipeak Curve measure the probability that a new location of x will lead to an improvement upon the incumbent $f(x^+)$ (with $x^+ = \arg\min_{x_i \in x_{1:t}} f(x_i)$ the best location so far). The mathematical expression would be:

$$\alpha_{PI}(x; \mathcal{D}_n) = \mathcal{P}(f(x^+) > f(x)) = \Phi(\frac{f(x^+) - \mu_n(x)}{\sigma_n(x)})$$

with $\Phi$ the standard normal cumulative distribution function.

Intuitively this approach has a major drawback, is that it still favours exploitation (extreme (really small) $\mu_n(x)$ is better) over exploration (extreme (really big) $\sigma_n(x)$ is worse). As explained before this would lead to the problem of local optimisation (which is not our goal). One trick to overcome this is to add another $\xi$ ($> 0$, trade-off parameter)into the function:

$$\alpha_{PI}(x; \mathcal{D}_n) = \mathcal{P}(f(x^+) + \xi > f(x)) = \Phi(\frac{f(x^+) + \xi - \mu_n(x)}{\sigma_n(x)})$$

TODO: add a figure to explain the mechanism better.

### 2.3.2 Information-based policies

TOWRITE

## 2.4 Infill Optimisation

””The infill optimizer searches for the point x which yields the best infill value. Unlike the original optimization problem on f, the optimization on the infill cri terion can be considered inexpensive. While this is still a black-box optimization problem, points can be evaluated more lavishly, and Jones et al. [3] propose a branch and bound algorithm for this task. mlrMBO defaults to a more generic ap proach, which we call focus search, outlined in Algorithm 1. It is able to handle numeric parameter spaces, categorical parameter spaces, as well as mixed and hierarchical spaces. The algorithm starts with a large random design from which all points are evaluated by the surrogate regression model to determine the most promising point. Next, focus search shrinks the search space around the best point and samples new random points for the now focused search space. The shrinkage of search space is iterated niters times. The complete procedure can be restarted nrestart times to avoid local optima. Finally the best point over all restarts and iterations is returned. Evolutionary algorithms like CMA-ES [26] or custom user-defined optimizers can be selected alternatively.”

**Algorithm 1** Infill Optimization: Focus Search.

```
Require: infill criterion c :$\mathcal{X} \to \mathbf{R}$, control parameters $nrestart$, $niters$, $n_{points}$
Ensure: $y = x^n$
$y \leftarrow 1$
$X \leftarrow x$
$N \leftarrow n$
while $N \neq 0$ do
    if $N$ is even then
        $X \leftarrow X \times X$
        $N \leftarrow \frac{N}{2}$                    ▷ This is a comment
    else if $N$ is odd then
        $y \leftarrow y \times X$
        $N \leftarrow N - 1$
    end if
end while
```

## 2.5 SMBO in Hyperparemeter Optimisation

TOWRITE

| **Acts** | **States of nature** | |
|---|---|---|
| | good | rotten |
| Break into bowl | six egg omlette | no omlette/<br>six eggs wasted |
| Break into saucer | six egg omlette<br>+ one extra thing to clean | 5 omlette<br>+ one extra thing to clean |
| Throw away | five egg omlette + one wasted egg | five egg omlette |

Table 1: Savage's Omlette Dilemma

# 3 Fundamental principles of Classical Decision Theory

## 3.1 The classical decision problem

### 3.1.1 The basic model

Perhaps the most used example for introducing fundamentals of a classical decision problem is the Savage omelette problem (see *The foundations of statistics. By Leonard J. Savage, John Wiley & Sons, Inc., 1954, 294 pp* (1954)), a dilemma for one decision maker: one try to make an omelette; there are five eggs already broken into a bowl and an unbroken egg number six. One has three options: either break the sixth egg into the same bowl, break open it into another saucer, or throw it away without checking it. For more context, the first five eggs are not rotten, but there is a chance that the sixth egg is and the only way to find out is to break open the egg. Now if he opens the egg into the bowl then there is a chance he might ruin the whole 6 eggs if the sixth one is rotten; putting it in the saucer seems like a better solution, but now one has two things to clean up. It leaves us with throwing the egg away, which is not ideal considering wasting a whole perfectly good egg. The options leading to actions are called **acts**; the two conditions (good or rotten) of the sixth egg are called **states of nature**; and the results from after the decision is made are named **consequences**. **??** summarises the decision problem.

Depends on what the decision maker labels as huge **risk** (cleaning extra thing or wasting egg(s)), he will then come to own conclusion of what decision turns out best for him. In this thesis we only consider the basic **classical** decision problems where there are finite number of:

- acts to choose from
- states of nature to be aware of
- consequences (gain/utility or risk), each corresponds to the act and the state of nature, so e.g.: if there are three acts and three states then there will be $3 * 3 = 9$ consequences

If the decision maker has no empirical data or observed information about the **likelihood** of various outcomes or **states of nature**, we call this decision problem a **no data** decision problem, which mean that we will have to consider the problem of making decisions **under uncertainty** and **under ignorance**. That, once again, would mean that an agent knows about all the states of nature and their utilities, but no other information about these states of nature or prior probabilities about them are known (see Savage (1951); Vajda (1959)). From here we introduce a general definition of a classical decision problem (CDP) (also see Jansen (2015)):

A classical (no-data) decision problem (CDP) (in utility form) is a triplet:

$$U := (\mathbf{A}, \Theta, u(.))$$

with

- $\mathbf{A}$ a (finite) non-empty set of **acts**
- $\Theta$ a (finite) non-empty set of **states of nature**
- u(.) a mapping $(\mathbf{A} \times \Theta) \rightarrow \mathbf{R}$, $(a, \theta) \mapsto u(a, \theta)$; the output should also be a (finite) non-empty set of utilities.

**Remark**: these problems could also be defined in **loss** form (instead of **utility form**) by including l (loss) instead of u (utility) in the triplet, depending on whether the values of consequences should be interpreted as loss or utility.

When the problem U is finite (as in the example of Savage's dilemma), we can formulate the triplet as such: $\mathbf{A} = \{a_1, .., a_n\}$ and $\Theta = \{\theta_1, .., \theta_m\}$ and the utilities can be put on a table:

| u(a$_i, \theta_j$) | $\theta_1$ | ... | $\theta_m$ |
|---|---|---|---|
| a$_1$ | u(a$_1, \theta_1$) | | u(a$_1, \theta_m$) |
| $\vdots$ | $\vdots$ | ... | $\vdots$ |
| a$_n$ | u(a$_n, \theta_1$) | ... | u(a$_n, \theta_n$) |

Table 2: Table with all acts and states

Here, the entry (i,j) of the table equals the utility of the pair $(a_{i},_{j})$. The letters n and m always denote the cardinalities of the sets A and Θ respectively.

## 3.2 Different criteria of Decision Theory

Having stated the essential components defining a CDP, the question now is how one should make their decision about which action should be chosen? We start with a special case of CDP, where it might happen that in a problem we have a clear winning strategy, where choosing to follow an action would lead to the best outcomes (in favour of the decision maker), regardless of states of nature; or the other way around where one action should not be chosen at all under any condition. This leads to the following definition (see Luce and Raiffa (1989), Jansen (2015)):

Let **A** be any CDP. An action $a \in$ **A** is said to be **inadmissible**, if there exists an action $a \in$ **A**, such that:

1. u($a^*$, $\theta$) $\geq u(a,\theta)\ \forall\ \theta \in \Theta$
2. u($a^*$, $\theta$) > u(a, $\theta$) $\exists\ \theta \in \Theta$

If that happens we call $a^*$ strictly dominates a and write $a^* \gg$ a or $a^* \succeq$ a. Table 3 illustrates an example of inadmissible action: $a_2$ strictly dominates $a_1$, which makes $a_1$ inadmissible. $a_3$ and $a_2$ on the other hand cannot be compare w.r.t. the relation $\gg$. In this case one will work with a reduced set of actions, consisting of only $a_2$ and $a_3$.

In most cases, however, there would be no inadmissable action: we need to come up with criteria that make all actions comparable meanwhile still being admissible. This motivates our next general definition (see Jansen (2015)):

Let U be a CDP, then we call any mapping a **criterion** such that:

$$\Phi : \mathbf{A} \rightarrow \mathbf{R}, a \mapsto \Phi(a)$$

An action $a^* \in$ **A** is **optimal** for the criterion $\Phi^U$, if $\Phi^U(a^*) \geq \Phi^U(a)$ holds for all a $\in$ **A**

| **Acts** | **States** | | |
|---|---|---|---|
| | $\theta_1$ | $\theta_2$ | $\theta_3$ |
| $a_1$ | 12 | 13 | 14 |
| $a_2$ | 14 | 15 | 16 |
| $a_3$ | 13 | 17 | 12 |

Table 3: Example of inadmissible action

Before we dive into the criteria for optimal decision making, let us mention one more relevant concept of decision theory: the different types of uncertainty. Jansen (2015) classifies into 2 types, depending on the nature of the **process generating** the states of nature:

- Type I: The realised state of nature is **independent** of the action chosen by the actor. The process generating the states of nature can be compared to an ideal lottery, as every state occurs with a fixed and known (classical) probability. The nature cannot influence the generation of the states.
- Type II: For every action the actor chooses, the nature will pick (one of the) state(s) minimising the actor's utility. In this case, the nature can fully influence the process generating the states of nature.

As these two sound very unrealistic and restrictive w.r.t. (non-academic) situations, Jansen (2015) also states two modifications:

- Type $I^*$: Exactly like in the type I situation, but the exact probability mass function on the set $\theta$ is assumed to be unknown to the actor (decision-maker). Based on the quality of information about the process generating the states, the decision-maker may be able to provide their best subjective estimate of the true probability distribution.
- Type $II^*$: Again, the nature acts as an antagonist to the actor. However, now the nature will not know exactly which action the decision maker will choose in order to pick the state minimising the actor's utility. Jansen (2015) provides the example of playing chess against a friend, where the friend will try to pick theleast favourable move possible, but he could not possibly know exactly which move you would go for after that.

Having the two types stated, we want to construct our criteria based on this belief about the type of uncertainty. Remark from Jansen (2015): it would be horrible to construct criteria for the optimality of decisions based on wrong beliefs about the true type of uncertainty. Our following discussion will base on the assumption that actor's belief and the true type of uncertainty coincide. For the sake of completeness it should also be mentioned that this thesis only dicusses aboout some of the most common criteria based on **cardinal utility**: on the assumption that the consequences (utilities) of the acts for the different states of nature (of the underlying utility table) can be ranked in a cardinal order. Beyond that there are also criteria based on ordinal utility (where the cardinality is not available/questionable) and criteria based on Preference Systems, where instead of an utility table the actors' preferences is represented by a preference system.

### 3.2.1 Optimal criteria under type $I/I^*$ uncertainty: Bernoulli-/Bayes-Actions

As the probability of each state is known (Type I), a straightforward criterion would be: an action is optimal iff it maximises the expected utility under the state of nature. With that in mind let us first define the so-called **expected ultility**. According to Augustin (2024), the **expected ultility** (of an action a w.r.t. $\pi$) of a data-free decision problem $(\mathbf{A}, \Theta, u(.))$ (in utility form) for each fixed a $\in \mathbf{A}$ can be expressed as:

$$\mathbf{E}_\pi(u(a)) := \int_\Theta u(a,\theta)\, d_\pi(\theta)$$

Important here are the conditions coming with the expression:

- $\pi$ a probability measure on $(\Theta, \sigma(\Theta))$ (probability for each state of nature to happen)
- $\sigma(\Theta)$ is an $\sigma$-algebra over $\Theta$ and $\theta \in \sigma(\boldsymbol{\Theta})$ for all $\theta \in \Theta$
- $u(a,\theta)$: $\Theta \to \mathbf{R}$, $\theta \mapsto u(a,\theta)$ is $\sigma(\Theta) - \mathcal{B} - \mathit{measurable}$) and an $\pi$- integrable mapping for all a $\in \mathbf{A}$

From that the **Bernoulli criterion** can be defined as:

$$\Phi(.,\pi) = \mathbf{A} \to \mathbf{R}, a \mapsto \Phi(a,\pi) = \mathbf{E}_\pi(u(a))$$

with our goal is to find the (one) action $a^*$ so that $\mathbf{E}_\pi(u(a^*, \theta)) \geq \mathbf{E}_\pi(u(a), \theta)$ for every a $\in \mathbf{A}$.

Under Type $I$ uncertainty, the exact probability measure $\pi$ is unknown, which makes the criterion above inapplicable - instead we will use the Bayes-criterion: using a prior-distribution $\xi$ estimating $\pi$. In this thesis we will not dig into more details, if one wants to know more please refer to Augustin (2002) and Jansen (2015).

### 3.2.2 Optimal criteria under type $II/II^*$ uncertainty: Maximin-Actions

Under Type $II$ the decision maker should always expect the worst state of nature possible (that minimise their utility (pessimistic)) . Looking this way, one should opt for the action that maximises the utility in the worts-case-scenario. This criterion is widely known as the Maximin-principle (or Wald-rule) (see Wald (1951) for further details), and could be mathematically defined as such:

$$\Phi(a) = \mathbf{A} \to \mathbf{R}, a \mapsto \Phi(a) = \inf_{\theta \in \Theta} u(a, \theta)$$

If the decision maker concerns about loss (risk) more than how much they could earn through the action, then the criterion could be defined as:

$$\Phi(a) = \mathbf{A} \to \mathbf{R}, a \mapsto \Phi(a) = \sup_{\theta \in \Theta} l(a, \theta)$$

with the logic that $\sup_{\theta \in \Theta} l(a, \theta)$ equivalent to the case of $\inf_{\theta \in \Theta} u(a, \theta)$. As our goal is to find an action $a^*$ so that $\inf_{\theta \in \Theta} u(a*, \theta) \geq \inf_{\theta \in \Theta} u(a, \theta)$ for every $a \in \mathbf{A}$, one could express the criterion in this way: an action $a^* \in \mathbf{A}$ is optimal iff:

$$\Phi(a^*) = \max_{a \in \mathbf{A}}(\min_{\theta \in \Theta}(u(a, \pi)))$$

hence the name of the criterion. It is worth to mention that even under type $II^*$ uncertainty, this **maximin** strategy will still be the best approach - while more details on this will not be discussed further, one can read Jansen (2015) for more information.

TODO: Bsp für minimax

Unfortunately, the minimax regret [risk] criterion has several drawbacks. First, it has never been clearly demonstrated that differences in utility do in fact measure what one maycall regret [risk]. In other words, it is not clear that the "regret" of going from state of utility 5 to a state of utility 3 is equivalent in some sense to that of going from state of utility 11 to one of utility 9. Secondly, one may construct examples where an arbitrarily small advantage in onestate of nature outweighs a considerable advantagein another state. Such examples tend to produce the samefeelings of uneasiness which led manyto object to the [maximin utility] criterion. A third objection which the author considers very serious is the following. In some examples the minimax regret criterion mayselect a strategy [act] Az among the available strategies? Ai, A2, A3, and Ag. On the other hand,if for some reason A, is made unavailable, the minimaxregretcriterion will select Az among Aj, Ag, and A3. The authorfeels that for a reasonable criterion the presence of an undesirable strategy A4 should not have an influence on the choice among the remaining strategies

TODO: add a Bsp.

Apart from these pessimistic point of view of the outcome state of nature (that always minimises actor's gain / maximises actor's loss (risk/regret), there exist a criterion where we assume the best outcome in favour of the actor: the **maximax** (Max-Max) criterion. Knowing the two criteria above, it is quit intuitive for one to come up with the expression for this optimal strategy: an action $a*$ is optimal iff:

$$\Phi(a^*) = \max_{a \in \mathbf{A}} (\max_{\theta \in \Theta} (u(a, \pi)))$$

While this criterion should be considered too optimistic for most of the cases in real life, it is still included here in this thesis as a foundation for our next criterion as a combination of the aforementioned criteria: the **Hurwicz** criterion. First presented in a paper in 1951, it introduces an approach for balancing optimism and pessimism in decision making **under uncertainty** (see Hurwicz (1951) and Hurwicz (2024)):

$$\Phi(a) = \mathbf{A} \to \mathbf{R}, a \mapsto \Phi(a) = \alpha \max_{\theta \in \Theta} u(a, \theta) + (1 - \alpha) \min_{\theta \in \Theta} u(a, \theta)$$

with $\alpha \in [0, 1]$ the **optimism** parameter and $|\Theta| < \infty$.

### 3.2.3 Combination of Type I and II: The Hodges Lehmann criterion

Instead of having a clear case of under which uncertainty type we are working with, a lot of cases will be a mixture of those two extremes: a criterion considering both at the same time would be needed, and in fact there are many different proposed criteria in the literature, however, this work will focus only on one of them: The criterion of Hodges Lehmann from 1952 (see Hodges and Lehmann (1952)). The criterion could be expressed as such:

$$\Phi(a) = \mathbf{A} \to \mathbf{R}, a \mapsto \Phi(a) = \alpha \mathbf{E}_\pi(u(a)) + (1 - \alpha) \min_{\theta \in \Theta} u(a, \theta)$$

with $\alpha \in [0, 1]$ the **confidence** parameter or **trade-off** parameter.

The idea is quite intuitive: Any type of uncertainty can be viewed as a trade-off between type I and type II uncertainty. The optimal criterion under type I uncertainty is assumed to be the Bernoulli-criterion (here the first addend in the formula), the optimal criterion under type II uncertainty is assumed to be the Maximin criterion (the second addend in the formula). If $\alpha$ was closer to 1, the more the optimal criterion would agree with the Bernoulli-criterion; vice versa the optimal criterion would align with the Maximin-criterion when the $\alpha$ got closer to 0. The question now is how one can be certain which value should $\alpha$ take/ which type of uncertainty is underlying the CDP of interest? In reality one can only have vague guesses that it is neither strict type I nor strict type II

uncertainty - the best solution would be to fix $\alpha$ best possible and act as if the criterion perfectly fits the underlying uncertainty type (Jansen (2015)).
TODO: essempio

# 4 Alternative Acquisition Functions applying Decision Theory

As the goal of this thesis is to discuss the mechanism underlying the method of Bayesian Optimisation from the view of a decision theoretician, this chapter aims to introduce some new infill functions applying the optimally criteria discussed in the previous chapter.

**Revision**: Before we deep dive into these infill functions, we wish to revise our goal of optimisation: in Chapter 2.1 it is stated: we need to find x in a sample space of $\mathcal{X}$, so that our unknown function f(x) get its optimal value (in context of this thesis the minimum value), so we want to find the one $x^*$ so that:

$$x^* = \arg\min_{x \in \mathcal{X}} f(x)$$

with $\mathcal{X}$ a d-dimensional space. Our approach is that we get our initial values and fit a surrogate model (Gaussian Process Regression), from that we receive our parameter: the posterior mean response $\mu$ and the posterior standard deviation $\sigma$. With these two and using a certain acquisition function we could decide which location in the $\mathcal{X}$ space might contain the optimal value we are looking for. Since we want to find the optimal value with the least "digging" through the search space $\mathcal{X}$ as possible (because of cost-sensitivity), our approach is that we make all future decisions optimally a.k.a. the Bellman's principle of optimisation (Bellman (1952)). That woule mean that our acquisition function should opt for the best possible candidate it could find. This approach is called **myopic**, as we only look one step ahead at a time into the future. Other approach considering multi-steps ahead at a time into the future are also proven to be useful (see Yue and Kontar (2020), Di Fiore and Mainini (2024), Jiang et al. (2020)), but we don't need to talk about them here any further.

Now in order to come up with new infill function applying the optimal criteria from decision theory, we need to determine how the relevant concepts in two fields (Decision Theory and Bayesian Optimisation) relate to each other. We start with the concept of **act** (or action) from Decision Theory: The actor makes a decision to **choose an act** from a set of infinite possible acts - in the case of black-box function that would be **choosing an x** from a set $\mathcal{X}$ of all possible values for x. Then we have states of natures - this one is a bit trickier - one would have to really understand how the Gaussian Process really works. Figure 3 gives a picture of how the nature would affect on its outcome states.

It could be understood that each state of nature would equal to one different regression function in Figure 3. As the possible functions are finite, our states of nature are also finite. Since we cannot be sure of the exact regression function underlie the our black-box function, there would always be uncertainty, which is indicated by the confidence bound with standard deviation $\sigma$ (Figure 5).

From the confidence bound (CB) we now have an understanding of how the states of nature look like (where the limits are). For example when one choose $x_1$ (the action), the states of nature would result in a range of evaluation values $[\mu(x_1) - \sigma(x_1),\, \mu(x_1) + \sigma(x_1)]$,

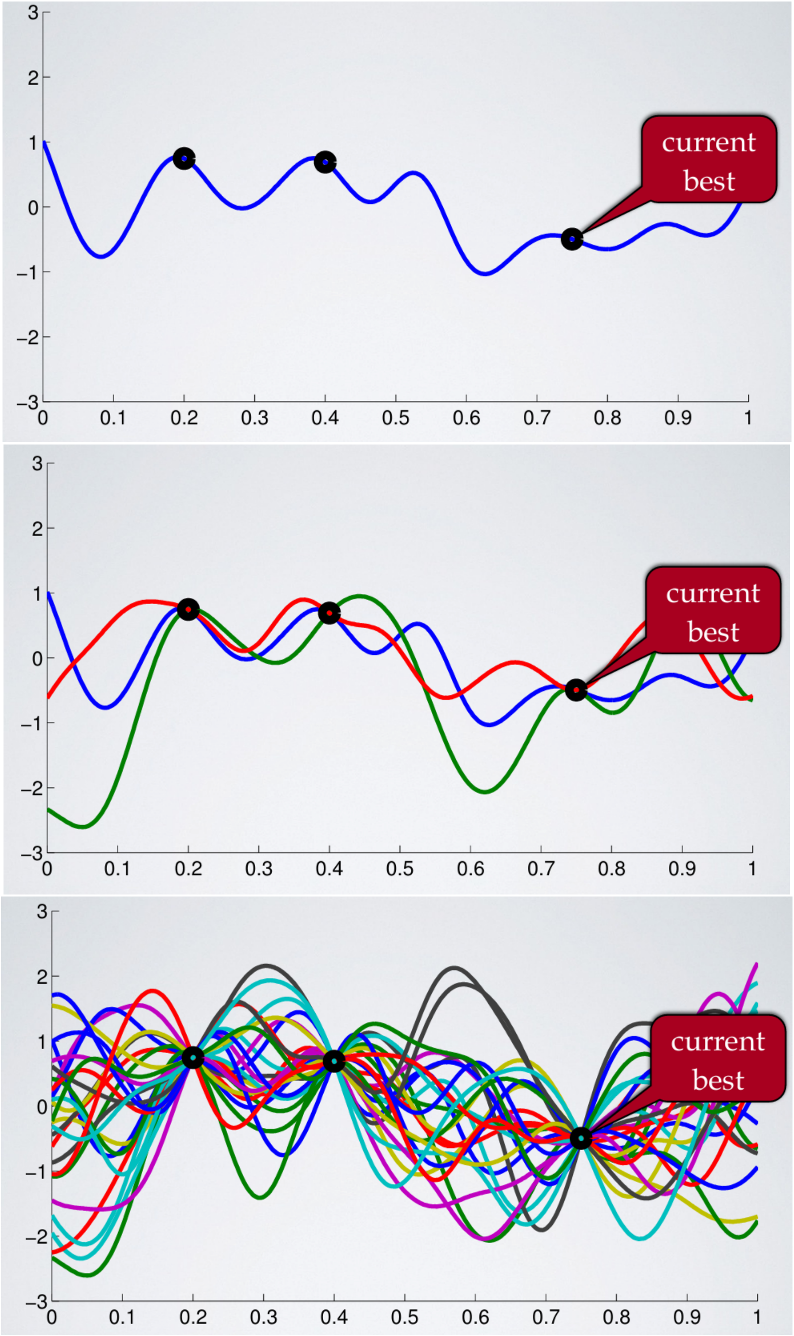

Figure 3: Given the first three initial points from the sample space we could fit a finite amount of regression function. This is exactly why fitting a simple regression function would not work too well. Source: Adams (2018)

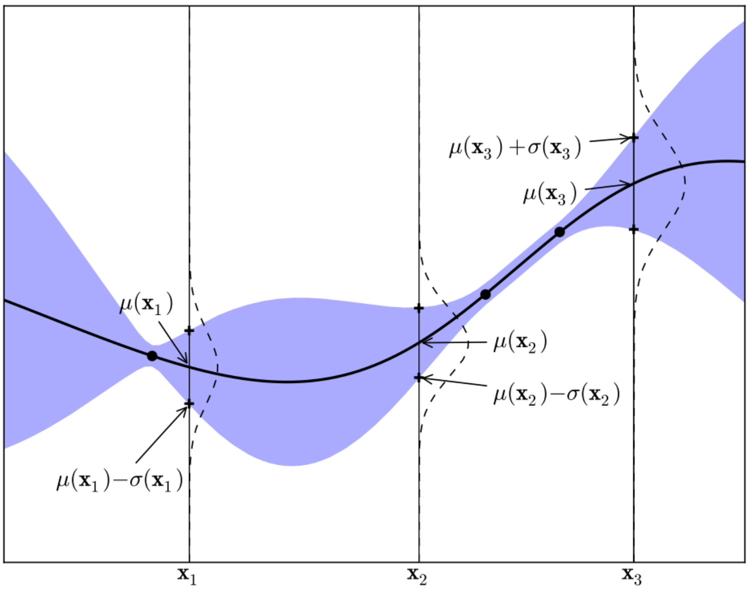

Figure 4: From the first three initial points we fit a Gaussian Process Model, which comes with a confidence bound (in purple). Source: Mayer (2017)

which could be interpreted here as utility (gain). In our case of searching for the minimum, if we choose $x_1$ as our next "act", we "gain" the most if the state of nature results in the evaluation on the border of the lower bound, and we "gain" least if we "land" on the border of the upper bound during the evaluation.
ref

## 4.1 Maximax criterion

As a revision, the **Maximax** criterion refers to the case when the decision maker is very optimistic that the best outcome would always happen, regardless of action they choose. With that mindset the actor would neglect the bad cases and only focus on picking the best case scenario option. For example, in the case of choosing one from the three options $x_1, x_2$ and $x_3$ in Figure 5, one would make decision based solely on the evaluation on lower border of the lower confidence bound (with the value $\mu(x_i)-\sigma(x_i)$ with i $\in 1, 2, 3$). If these were the only 3 options to pick from, one would go with $x_1$ as the value of $\mu(x_1) - \sigma(x_1)$ is the lowest out of the three.

From that we define our first alternative infill criterion using the maximax strategy: $x^*$ should be chosen as the next location to evaluate iff:

$$x^* = \arg\min_{x \in \mathcal{X}} (\mu(x) - \lambda\sigma(x))$$

One should recognise now that this is the exact mathematical definition of the acquisition function using the upper confidence bound criterion. We could also controll the trade-off between exploration (higher $\sigma$) and exploitation (lower $\mu$) by including $\lambda > 0$ as a parameter. We will howerver set by default $\lambda = 1$.

## 4.2 Maximin criterion

As opposed to the cases using the maximax criterion, using the Maximin criterion (as an acquisition function in Bayesian Optimisation) shows that the decision maker is pessimistic with the outcome, that they always picture the worst case would happen, and aim to maximise the result in that case (here getting the largest value of $y_i$ in any $x_i$ location, which is when the possible value of $y_i$ lands on the upper confidence bound). Therefore the objective of problem could be expressed as:

$$x^* = \arg\min_{x \in \mathcal{X}}(\mu(x) + \lambda\sigma(x))$$

Here the decision maker consider the worst possible evaluation at each candidate location $x_i$, then pick out the minimum out of these evaluation $(\mu(x_i) + \lambda\sigma(x_i))$.

## 4.3 Hurwicz criterion

From the idea of Hurwicz presented in Chapter 3, we are here firstly trying to find the compromise between an optimistic decision (when the $y_i$ value lands on the lower confidence bound) and a pessimistic decision (when the $y_i$ value lands on the upper confidence bound). The $\alpha$ (a.k.a. the **coefficient of realism**) takes value in the bounded interval between 0 and 1 and shows how optimistic we are that the best result of $y_i$ will happen when we decide to evaluate the point $x_i$. The mathematical expression would hence be:

$$x^* = \arg\min_{x \in \mathcal{X}}(\alpha(\mu(x) - \lambda\sigma(x)) + (1 - \alpha)(\mu(x) + \lambda\sigma(x))$$

## 4.4 Minimax Regret criterion

In Bayesian Optimisation, a state of nature corresponds to a specific realization of the unknown function f(x) that the surrogate model is attempting to approximate. The surrogate model, such as a Gaussian Process (GP), provides a probability distribution over these states. As another revision: for each x, the GPR gives:

- A predictive mean $\mu(x)$ , which is the expected value of f(x)
- A predictive standard deviation $\sigma(x)$, which quantifies uncertainty about f(x)

Each state of nature corresponds to one possible realizstion of f(x) drawn from the GP posterior distribution.

The risk measures the performance of a specific decision x relative to the unknown states of nature. In the minimax context, we calculate risk as such: Regret for choosing x under a specific state of nature is the difference between:

- The best possible gain (across all decisions and given the state of nature).
- The gain achieved by x.

In terms of the surrogate model: The best possible gain is approximated as: $\min_{x \in \mathcal{X} f(x)}$ (as we try to minimise the target function). For that the true regret (risk) for a decision x in a specific state of nature f(x) should be:

$$R(x, f) = f(x) - f(x^*)$$

with $x^*$ the location where f gets its minimum value in this iteration. From this we want to consider for **each decision x** (in this iteration) the **greatest risk** that is available, then from those potential risks (each respective to one decision x) we pick out the one decision that has the smallest risk out of all. With mathematical expression one could write, that the decision $x'$ is optimal for this criterion iff:

$$x' = \arg\min_{x \in \mathcal{X}} \max_{f} (f(x) - f(x^*))$$

This ensures that even in the worst-case realisation of f, the regret of choosing the optimal candidate x' is as small as possible.

But as we do not know this f(x) for each particular state of nature,, we rely on the GP posterior (mean $\mu(x)$ and variance $\sigma(x)^2$ to approximate regret. First we approximate the true minimum:

$$\min_{x \in \mathcal{X}} f(x) = \min_{x \in \mathcal{X}} \mu(x) = \hat{f}_{min}$$

Then the approximate Regret for a decision x should be:

$$R(x) = \mu(x) - \hat{f}_{min}$$

To account for uncertainty in the GP model, we adjust the regret using the variance:

$$R(x) = \mu(x) + \lambda\sigma(x) - \min_{x \in \mathcal{X}} (\mu(x) + \lambda\sigma(x))$$

where$\lambda > 0$ is a parameter controlling the tradeoff between exploration (higher $\sigma$) and exploitation (lower $\mu$). The criterion should be then written as:

$$x' = \arg\min_{x \in \mathcal{X}} \max_{f} R(x)$$

## 4.5 Bayes criterion

The main objective is to choose the action with the best expected utility (here is the minimum). As our surrogate model is the Gaussian process, we could directly extract the expected value of $y_i$ at any location of $x_i$ ($\mu(x_i)$). We come to the expression, which is the same as the criterion of reinstating pure mean $\mu(x)$ minimization (pure exploitation):

$$x^* = \arg\min_{x \in \mathcal{X}} \mu(x)$$

## 4.6 Hodges-Lehmann criterion

This approach is a combination of the aforementioned Maximin Criterion and the Bayes Criterion. The weighting of the trade-off between these two extreme criteria can be controlled by a trade-off parameter $\alpha \in [0, 1]$:

$$x^* = \alpha(\arg\min_{x \in \mathcal{X}} \mu(x)) + (1 - \alpha)(\arg\min_{x \in \mathcal{X}}(\mu(x) + \lambda\sigma(x)))$$

$$= \arg\min_{x \in \mathcal{X}}(\alpha\mu(x) + (1 - \alpha)(\mu(x) + \lambda\sigma(x))$$

$$= \arg\min_{x \in \mathcal{X}}(\mu(x) + (1 - \alpha)(\mu(x) + \lambda\sigma(x))$$

| Test functions | Formula | Properties | Domain | Minimum Location |
|---|---|---|---|---|
| Deflected Corrugated Spring | f(x) $= 0.1\sum_{i=1}^{d}(x_i-\alpha)^2 - cos(K\sqrt{\sum_{i=1}^{d}(x_i-\alpha)^2})$<br>$\alpha = K = 5$ by default | highly multimodal, symmetric away from the next | [0,2$\alpha$] | $x_i^* = \alpha$<br>and $f_{min} = -1$ |
| Schwefel | f(x) $= \sum_{i=1}^{d} -x_i sin(\sqrt{\lvert x_i \rvert})$ | highly multimodal, global optimum far away from the next best local optimum | [-500,500] | $x_i^* = 420.9687$<br>and $f_{min} = 0$ |
| Rosenbrock | f(x)$= \sum_{i=1}^{d-1} 100(x_{i+1} - x_i^2)^2 + (1-x_i)^2$ | unimodel, non-convex, differentiable | [-30,30] | $x_i^* = 1$<br>and $f_{min} = 0$ |
| Ackley | f(x) $= -20exp(-0.2\sqrt{\frac{1}{d}\sum_{i=1}^{d} x_i}) - \exp(\frac{1}{d}cos(2_i))$ | differentiable, multimodal | [-32.8,32.8] | $x_i^* = 420.9687$<br>and $f_{min} = 0$ |

Table 4: Test functions used for benchmarking and their properties

# 5 Experiments

## 5.1 Experiments with synthetic test functions

We are now at the last part of this thesis, where we are going to benchmark the new infill functions with different **synthetic** testing functions (in total of 10 objective functions) from the package **smoof** (see Bossek (2017)) to get a first understanding of how they would perform. Although we know the explicit mathematical form of all these testing functions and where exact their minima are located (details will be explained later on), we still consider them as black box function for the sake of our test (i.e. we are only able to get the output values y when we input some values of x, any more details of the objective function than that should be considered beyond our knowledge). Now for each of these target functions, we run the BO algorithm in order to find the **target global minimum**, each time with different infill functions - for each pair of testing and infill function we will run the test 40 times, then measure and compare the best result we receive and the running time needed to reach that decision. For the whole benchmark the package for implementing BO strategies will be **mlrMBO** (see Bischl et al. (2018)).

### 5.1.1 Problem design

With problem design we mean the design of the objective functions. In this thesis, we will only focus on the Model-Based Single-Objective Optimisation, which involves optimising (minimising in our case) a single scalar function, often representing a single goal or criterion.

For this benchmark we pick three 2-dimensional, three 4-dimensional and four 6-dimensional target function, which are all continuous, and single-objective: Deflected Corrugated Spring, Schwefel, Rosenbrock and Ackley. All are defined in the R-package **smoof** (see Bossek (2017)) and have been subject to optimisation benchmarks previously. The Table 4 shows the properties of each of the 10 objective functions used in this benchmark.

### 5.1.2 Algorithm design

The algorithms here are the BO strategies, which in our thesis only differs from each other in their acquisition functions. Figure 5 demonstrates a common routine of a typical BO algorithm. We start our algorithm with collecting initial sample points from the black box target function.

**Initial sampling set**: Sampling **randomly** is a plausible approach for this, but that would be inefficient as we might only collect all the points from a small area in the sample space. Since the evaluations are considered costly, we want to find a way to sample the least amount of points as possible while still capturing the essence of the black box function. For that reason we will generate design for each problem by using a space-filling maximin Latin Hypercube Design (see McKay et al. (1979)). We then set the size of the design to be 5 times the number of the black-box function's parameters - if the problem is 2-dimensional then there will be 10 initial sampled points.

**Definition of the surrogate regression model**: Till this point it is clear that the our first choice regarding to choosing a surrogate model is GPR. It is also a rational choice such that if the sample space $\mathcal{X} \in \mathbf{R}^d$, then Kriging (a.k.a. GPR) (see Jones et al. (1998)) would be recommended and provides state-of-the-art performance. In our test we pick a fixed kernel the Matérn 3/2 which is mentioned in Chapter 2.2.2 and holds the form :

$$k_{Mat\acute{e}rn3}(\mathbf{x}, \mathbf{x}') = \theta_0^2 \exp\left(-\sqrt{3}\,r\right)\left(1 + \sqrt{3}\,r\right)$$

More details on this kernel could be found here Heuvelink (2000).

**Definition of the infill function**: as mentioned above we will want to put our new infill function in Chaper 4 into use here. Besides our alternative functions, we will also benchmark the popular **Expected Improvement** infill function to compare it with our new functions.

**Infill optimisation**: We use **focus search** (mentioned in Chapter 2.4) to optimise the infill functions (getting the next evaluation point), and keep all the parameters in default settings as in the package **mlrMBO**: $n_{restarts} = 3$, $n_{iters} = 5$ and $n_{points} = 1000$.

### 5.1.3 Execution of Experiments

Every experiment terminates if a total number of 200 function evaluations has been reached and returns the optimal configuration, regardless of the problem's dimension. Each experiment is repeated 40 times, as mentioned earlier.

### 5.1.4 Evaluation of Results

Besides the quality of the solution, runtime, and computational overhead should also be considered. The timings for a complete optimization run in minutes are listed in Table 1. Note that we are basically measuring the overhead of the optimization algorithms, as the synthetic test functions are evaluated in microseconds.

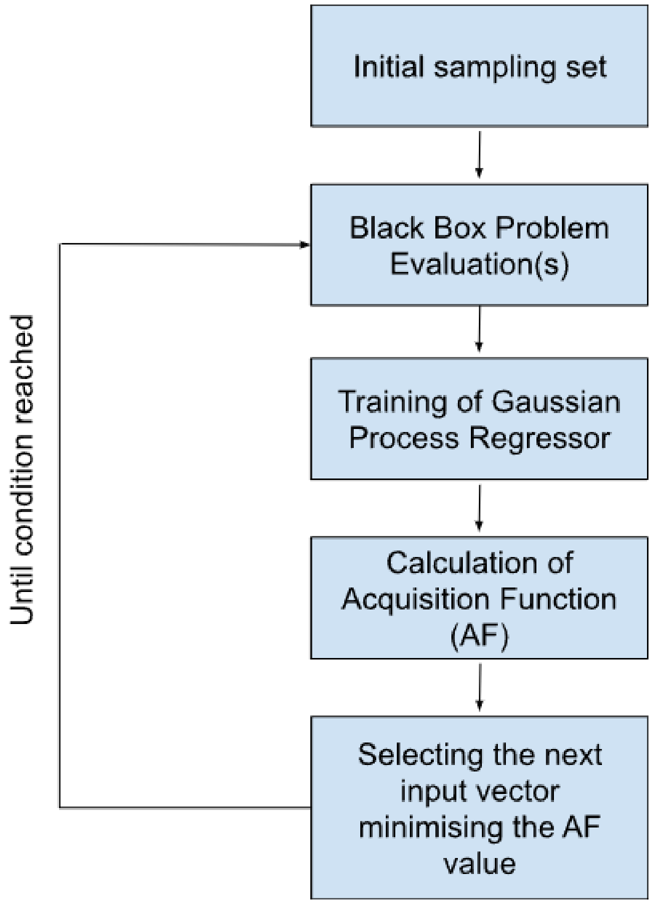
Initial sampling set
Black Box Problem Evaluation(s)
Training of Gaussian Process Regressor
Calculation of Acquisition Function (AF)
Selecting the next input vector minimising the AF value
Until condition reached

Figure 5: Basic routine of an BO algorithm. Source: paretos (2021)

## 5.2 Results and discussions

# 6 Conclusion and discussion

A concise summary of contents and results

### Future Work

# A Appendix

Additional material goes here

# B Electronic appendix

All of the code produced in this work can be found on:
https://github.com/DucAnhValentinoNguyen/dtMBO

# References

Adams, R. P. (2018). A tutorial on bayesian optimization for machine learning.
**URL:** *https:www.cs.toronto.edu/ rgrosse/courses/csc411*$_f$*18/tutorials/tut8*$_a$*dams*$_s$*lides.pdf*

Augustin, T. (2002). Expected utility within a generalized concept of probability — a comprehensive framework for decision making under ambiguity, *Statistical Papers* **43**(1): 5–22.
**URL:** *https://EconPapers.repec.org/RePEc:spr:stpapr:v:43:y:2002:i:1:p:5-22*

Augustin, T. (2024). Lecture desicion theory, chapter bayes decisions in the 'virtual risk situation.
**URL:** *https://moodle.lmu.de/mod/resource/view.php?id=1887619*

Bellman, R. (1952). On the theory of dynamic programming, *Proceedings of the National Academy of Sciences* **38**(8): 716–719.
**URL:** *https://www.pnas.org/doi/abs/10.1073/pnas.38.8.716*

Betrò, B. (1991). Bayesian methods in global optimization, *Journal of Global Optimization* **1**: 1–14.
**URL:** *https://api.semanticscholar.org/CorpusID:19324834*

Bischl, B., Richter, J., Bossek, J., Horn, D., Thomas, J. and Lang, M. (2018). mlrmbo: A modular framework for model-based optimization of expensive black-box functions.
**URL:** *https://arxiv.org/abs/1703.03373*

Bossek, J. (2017). smoof: Single- and multi-objective optimization test functions, *The R Journal* .
**URL:** *https://journal.r-project.org/archive/2017/RJ-2017-004/index.html*

Brochu, E., Cora, V. M. and de Freitas, N. (2010). A tutorial on bayesian optimization of expensive cost functions, with application to active user modeling and hierarchical reinforcement learning.
**URL:** *https://arxiv.org/abs/1012.2599*

Di Fiore, F. and Mainini, L. (2024). Nm2-bo: Non-myopic multifidelity bayesian optimization, *Knowledge-Based Systems* **299**: 111959.
**URL:** *http://dx.doi.org/10.1016/j.knosys.2024.111959*

Frazier, P. I. (2018). A tutorial on bayesian optimization.
**URL:** *https://arxiv.org/abs/1807.02811*

Guo, J., Zan, X., Wang, L., Lei, L., Ou, C. and Bai, S. (2023). A random forest regression with bayesian optimization-based method for fatigue strength prediction of ferrous alloys, *Engineering Fracture Mechanics* **293**: 109714.
**URL:** *https://www.sciencedirect.com/science/article/pii/S0013794423006720*

Heuvelink, G. (2000). Interpolation of spatial data: Some theory for kriging: M.l. stein, springer, new york, 1999. hardcover, 247 pp., us49.95,$isbn0 - 387 - 98629 - 4, Geoderma\mathbf{96}:$ 153–154.

Hodges, J. L. and Lehmann, E. L. (1952). The use of previous experience in reaching statistical decisions, *Annals of Mathematical Statistics* **23**: 47–58.
**URL:** *https://api.semanticscholar.org/CorpusID:119870153*

Hurwicz, L. (1951). The generalised bayes-minimax principle: A criterion for decision-making under uncertainty.
**URL:** *https://cowles.yale.edu/sites/default/files/2024-03/s-0355.pdf*

Hurwicz, L. (2024). The hurwicz criterion.
**URL:** *https://www.leonidhurwicz.org/hurwicz-criterion/*

Hutter, F., Hoos, H. H. and Leyton-Brown, K. (2011). Sequential model-based optimization for general algorithm configuration, *in* C. A. C. Coello (ed.), *Learning and Intelligent Optimization*, Springer Berlin Heidelberg, Berlin, Heidelberg, pp. 507–523.

Jansen, C. (2015). *Decision making under partial information using precise and imprecise probabilistic models*, PhD thesis.
**URL:** *http://nbn-resolving.de/urn/resolver.pl?urn=nbn:de:bvb:19-epub-25591-1*

Jiang, S., Jiang, D., Balandat, M., Karrer, B., Gardner, J. and Garnett, R. (2020). Efficient nonmyopic bayesian optimization via one-shot multi-step trees, *in* H. Larochelle, M. Ranzato, R. Hadsell, M. Balcan and H. Lin (eds), *Advances in Neural Information Processing Systems*, Vol. 33, Curran Associates, Inc., pp. 18039–18049.
**URL:** *https://proceedings.neurips.cc/paper*$_f$*iles/paper/2020/file/d1d5923fc822531bbfd9d87d4760914*
*Paper.pdf*

Jones, D. (2001). A taxonomy of global optimization methods based on response surfaces, *J. of Global Optimization* **21**: 345–383.

Jones, D., Schonlau, M. and Welch, W. (1998). Efficient global optimization of expensive black-box functions, *Journal of Global Optimization* **13**: 455–492.

Kushner, H. J. (1964). A new method of locating the maximum point of an arbitrary multipeak curve in the presence of noise, *Journal of Basic Engineering* **86**: 97–106.
**URL:** *https://api.semanticscholar.org/CorpusID:62599010*

Lizotte, D., Wang, T., Bowling, M. and Schuurmans, D. (2007). Automatic gait optimization with gaussian process regression, *Proceedings of the Twentieth International Joint Conference on Artificial Intelligence (IJCAI)*, pp. 944–949.

Luce, R. and Raiffa, H. (1989). *Games and Decisions: Introduction and Critical Survey*, Dover Books on Mathematics, Dover Publications.
**URL:** *https://books.google.de/books?id=msC*$_h$*0wtCO8C*

Ma, Y., Cao, D. and Feng, J. (2023). Application of random forest algorithm with bayesian optimization to reservoir prediction: Application of random forest algorithm with bayesian optimization to reservoir prediction, *2023 International Conference on New Trends in Computational Intelligence (NTCI)*, Vol. 1, pp. 412–418.

Martinez-Cantin, R., de Freitas, N., Brochu, E., Castellanos, J. and Doucet, A. (2009). A Bayesian exploration-exploitation approach for optimal online sensing and planning with a visually guided mobile robot, *Autonomous Robots* **27**(2): 93–103.
**URL:** *https://doi.org/10.1007/s10514-009-9130-2*

Martinez-Cantin, R., Freitas, N., Doucet, A. and Castellanos, J. (2007). Active policy learning for robot planning and exploration under uncertainty.

Mayer, V. (2017). Alternative infill-kriterien für random forests in sequential model-based optimization.
**URL:** *http://nbn-resolving.de/urn/resolver.pl?urn=nbn:de:bvb:19-epub-38411-8*

McKay, M. D., Beckman, R. J. and Conover, W. J. (1979). A comparison of three methods for selecting values of input variables in the analysis of output from a computer code, *Technometrics* **21**(2): 239–245.
**URL:** *http://www.jstor.org/stable/1268522*

Mockus, J. (1994). Application of Bayesian approach to numerical methods of global and stochastic optimization, *Journal of Global Optimization* **4**(4): 347–365.
**URL:** *http://link.springer.com/10.1007/BF01099263*

Mockus, J., Tiesis, V. and Zilinskas, A. (2014). *The application of Bayesian methods for seeking the extremum*, Vol. 2, pp. 117–129.

paretos (2021). Bayesian optimization (bayes opt): Easy explanation of popular hyperparameter tuning method.
**URL:** *https://www.youtube.com/watch?v=M-NTkxfd7-8t=353s*

Rasmussen, C. E. and Williams, C. K. I. (2005). *Gaussian Processes for Machine Learning*, The MIT Press.
**URL:** *https://doi.org/10.7551/mitpress/3206.001.0001*

Savage, L. J. (1951). The theory of statistical decision, *Journal of the American Statistical Association* **46**(253): 55–67.
**URL:** *http://www.jstor.org/stable/2280094*

Shahriari, B., Swersky, K., Wang, Z., Adams, R. and De Freitas, N. (2016). Taking the human out of the loop: A review of bayesian optimization, *Proceedings of the IEEE* **104**(1): 148–175. Publisher Copyright: © 1963-2012 IEEE.

Snoek, J., Larochelle, H. and Adams, R. P. (2012). Practical bayesian optimization of machine learning algorithms.
**URL:** *https://arxiv.org/abs/1206.2944*

*The foundations of statistics. By Leonard J. Savage, John Wiley & Sons, Inc., 1954, 294 pp* (1954). *Naval Research Logistics Quarterly* **1**(3): 236–236.
**URL:** *https://onlinelibrary.wiley.com/doi/abs/10.1002/nav.3800010316*

Thompson, W. R. (1933). On the likelihood that one unknown probability exceeds another in view of the evidence of two samples, *Biometrika* **25**(3/4): 285–294.
**URL:** *http://www.jstor.org/stable/2332286*

Thornton, C., Hutter, F., Hoos, H. H. and Leyton-Brown, K. (2013). Auto-weka: Combined selection and hyperparameter optimization of classification algorithms.
**URL:** *https://arxiv.org/abs/1208.3719*

Vajda, S. (1959). Games and decisions. by r. duncan luce and howard raiffa. pp. xi, 509. 70s. 1957. (j wiley 38;amp; sons), *The Mathematical Gazette* **43**(344): 152–153.

Wald, A. (1951). Statistical decision functions.
**URL:** *https://api.semanticscholar.org/CorpusID:4175992*

Yue, X. and Kontar, R. A. (2020). Why non-myopic bayesian optimization is promising and how far should we look-ahead? a study via rollout, *in* S. Chiappa and R. Calandra (eds), *Proceedings of the Twenty Third International Conference on Artificial Intelligence and Statistics*, Vol. 108 of *Proceedings of Machine Learning Research*, PMLR, pp. 2808–2818.
**URL:** *https://proceedings.mlr.press/v108/yue20b.html*

# Declaration of authorship

I hereby declare that the report submitted is my own unaided work. All direct or indirect sources used are acknowledged as references. I am aware that the Thesis in digital form can be examined for the use of unauthorized aid and in order to determine whether the report as a whole or parts incorporated in it may be deemed as plagiarism. For the comparison of my work with existing sources I agree that it shall be entered in a database where it shall also remain after examination, to enable comparison with future Theses submitted. Further rights of reproduction and usage, however, are not granted here. This paper was not previously presented to another examination board and has not been published.

Munich, date

Name