Bayesian Optimization using Information Gain Acquisition Functions

by

Ruifan (Ricky) Wu

A thesis submitted in conformity with the requirements
for the degree of Bachelor of Applied Science

Department of Mechanical & Industrial Engineering
University of Toronto

Bayesian Optimization using Information Gain Acquisition Functions

Ruifan (Ricky) Wu
Bachelor of Applied Science

Department of Mechanical & Industrial Engineering
University of Toronto
2025

Abstract

Bayesian Optimization (BO) has emerged as a powerful framework for optimizing expensive black-box functions commonly encountered in machine learning, engineering design, and scientific experimentation. Central to BO is the use of acquisition functions that guide the selection of evaluation points by balancing exploration and exploitation. While improvement-based acquisition functions, such as Probability of Improvement (PI) and Expected Improvement (EI), have shown success in many applications, they often struggle in high-dimensional or multimodal settings.

This thesis focuses on entropy-based acquisition functions, which adopt an information-theoretic approach by explicitly aiming to reduce uncertainty about the location or the value of the global optimum. We provide a review of entropy-driven strategies, including Entropy Search (ES), Predictive Entropy Search (PES), and Max-value Entropy Search (MES), outlining their theoretical foundations and the approximation techniques used to enable tractable computation.

Through a series of numerical experiments, we compare the performance of entropy-based and improvement-based methods across various benchmark functions. Our results highlight the advantages of information gain acquisition functions in terms of sample efficiency and robustness, particularly in scenarios where function evaluations are extremely costly.

Acknowledgements

This thesis is dedicated to my mentors, family, and friends who have supported me throughout my undergraduate journey. It has been a challenging yet transformative experience, and it is your encouragement and guidance that have shaped who I am today.

To my thesis supervisor and friend, Professor Justin J. Beland, thank you for the opportunity to work on this fascinating research topic and for your guidance throughout the project. I am deeply grateful for your patience in teaching me, from core academic concepts to the finer details of scholarly writing. Beyond your academic mentorship, your kindness and emotional support have helped me overcome the pressure in my graduating year. I couldn't dream of a better mentor.

To my friends Boyan Liang, Yuye Huang, Zhixiang Huang, Yujia Shi, Erxin Yan, and Yiling Ge, thank you all for being such wonderful companions throughout my undergraduate life. Your emotional support has meant more to me than words can express. Just as you've always had faith in me, I have faith in each of you. May we all shine in our own paths in life.

Lastly, to my parents, Zhihui Wu and Xiaoai Zhao, thank you for everything. I truly feel lucky to have grown up in an environment filled with support, respect, and love,that I have the motivation to dream and the power to explore my life with confidence and courage.

Thanks to everyone I've met over the past five years—you've all made this a truly wonderful life journey. Thank you for being a part of my life.

Contents

List of Figures

Chapter 1

Introduction

1.1 Background

In many real-world optimization problems, evaluating a solution can be computationally expensive, time-consuming, or resource-intensive. This is especially true in domains such as automated machine learning, drug discovery, materials design, and robotics, where each evaluation may involve training complex models, running detailed simulations, or conducting physical experiments [1] [2] [3]. Traditional optimization methods, ranging from exhaustive grid searches to heuristic-driven strategies, often become impractical in these settings.

To address these challenges, there is a growing need for optimization strategies that can make the most of limited evaluations while intelligently navigating complex, uncertain landscapes. One such approach is BO, which has emerged as a powerful framework for optimizing black-box functions with high evaluation costs. Rather than relying on large volumes of random or structured sampling, BO strategically selects evaluation points based on a surrogate model that captures uncertainty and guides the search process.

Central to the effectiveness of BO is the use of acquisition functions, which determine where to evaluate next by balancing the trade-off between exploring uncertain regions of the search space and exploiting areas that appear promising. Improvmentment based acquisition strategies, such as those based on expected improvement or confidence bounds, have demonstrated success in many practical applications. However, they can be limited in complex settings, particularly in high-dimensional or multimodal problems, where model predictions become unreliable or biased toward local optima.

In response, information-theoretic acquisition functions have gained attention for their ability to prioritize evaluations that reduce uncertainty about the location of the optimum itself. Rather than focusing solely on short-term gains, these methods seek to improve the model's global understanding of the objective function, often leading to faster convergence and more efficient use of resources. This shift from improvement-based heuristics to uncertainty-reduction strategies has significant implications for applications constrained by budget, time, or experimental cost.

This thesis investigates the theoretical motivations and practical benefits of BO with information gain-based acquisition functions. By comparing these approaches with traditional improvement-based methods across a range of benchmarks, we aim to understand their relative performance in guiding sample-efficient optimization.

1.2 Research Objective

The objectives of this thesis project are

- Construct entropy-based acquisition functions (i.e., Entropy Search, Predictive Entropy Search and Max Value Entropy Search) that can be used in a BO setting where we use the statistics from a GP posterior distribution to compute the search policy for ES, PES and/or MES.
- Compare the performance of these acquisition functions in a BO setting against other common acquisition functions such as the probability of improvement, expected improvement, lower confidence bound and/or Thompson sampling.

1.3 Thesis Layout

The remainder of this thesis is organized as follows:

In Chapter 2, we aim to review background information on the Gaussian Process(GP), BO, and based acquisition functions within BO Settings.

In Chapter 3, we aim to review background information on the Entropy Based Search methods.

In Chapter 4, we aim to carry out numerical studies that explain the experiment motivation, experiment setting, and the analysis on the experiment results.

Chapter 2

Literature Review

In this chapter we reviewed the BO framework, which employs a GP as a surrogate model and utilizes an acquisition function to determine the next sampling point. A detailed description of GP can be found in text by Rasmusen and Williams [4], while for BO the reader can refer to the text by Garnett et al. [5] for a comprehensive explanation.

2.1 Gaussian Process Overview

GPs are a powerful and widely used approach in BO for constructing surrogate models of hidden objective functions [4]. More broadly, GPs offer a probabilistic framework that seamlessly integrates with Bayesian Inference, allowing the model to make predictions while quantifying uncertainty. This makes them particularly valuable in scenarios where evaluating the true objective function is costly and only a limited number of observations are available.

This section introduces the reader to several relevant aspects of GP with the Function Space view of GP mentioned.

Formally, a GP is defined as a collection of random variables, where any finite subset follows a joint Gaussian distribution. Rather than specifying a function explicitly, a GP characterizes a distribution over functions, enabling a probabilistic approach to modeling unknown functions [4].

The GP model is completely specified by its mean function and covariance function. We define the mean function m(x) and the covariance function $k(\mathbf{x}, \mathbf{x}')$ of a real process f(x) as

$$m(\mathbf{x}) = \mathbb{E}[f(\mathbf{x})], \tag{2.1}$$

$$k(\mathbf{x}, \mathbf{x}') = \mathbb{E}[(f(\mathbf{x}) - m(\mathbf{x}))(f(\mathbf{x}') - m(\mathbf{x}'))], \tag{2.2}$$

We often use a compact notation of a GP with mean function m and covariance function f(x,x')

$$f(\mathbf{x}) \sim \mathcal{GP}(m, k) \tag{2.3}$$

Normally, for notational simplicity, we take the mean function to be zero, and when the mean function is non-zero, the GP can be rewritten as the offset with a mean function of $\hat{f} - m$

2.1.1 Gaussian Process Prior

In a GP Regression, we wish to use a GP as a surrogate and realization of the true objective function. For the default model used to approximate the hidden objective function, we typically assume a mean function of zero. The prior kernel (covariance) function captures the knowledge and assumptions we have regarding the hidden objective function.

Squared Exponential (RBF) Kernel

The squared exponential (SE) kernel, also known as the Radial Basis Function (RBF) or Gaussian kernel, defines the covariance between two input points $\mathbf{x}$ and $\mathbf{x}'$ in GPs [4], which takes the form of

$$k_{\mathrm{SE}}(\mathbf{x}, \mathbf{x}') = \sigma_f^2 \exp\left(-\frac{\|\mathbf{x} - \mathbf{x}'\|^2}{2\ell^2}\right) \tag{2.4}$$

The **length-scale** (ℓ) determines how quickly the function values change with respect to the input; a larger ℓ leads to smoother, more slowly varying functions, whereas a smaller ℓ allows the function to exhibit rapid, local fluctuations. On the other hand, the **signal variance** (σ_f^2) governs the overall magnitude of variation in the function's output, indicating how far the function values are expected to deviate from the mean. Together, these hyperparameters play a critical role in shaping the behavior of the function modeled by a GP, as shown in Figure 2.1.

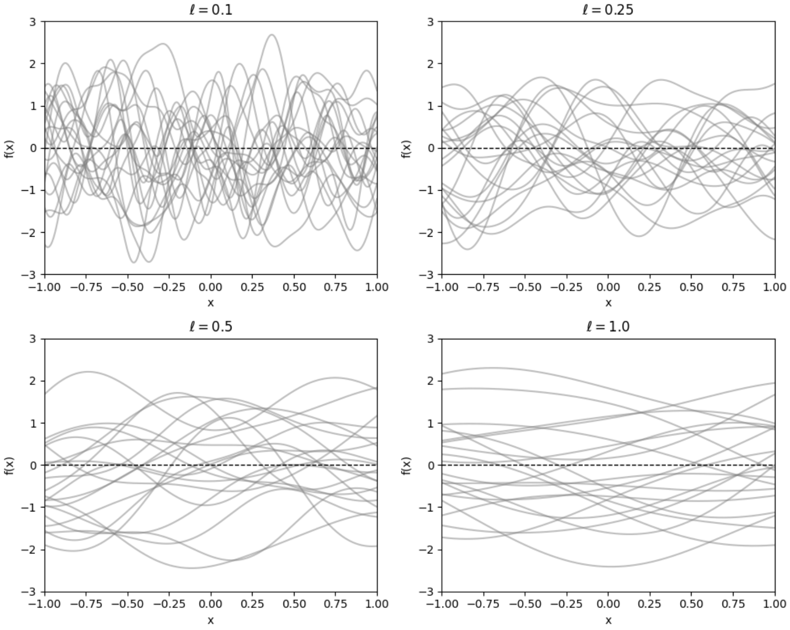


Figure 2.1: Samples drawn from a GP prior using the Squared Exponential (SE) kernel with varying length-scale values ℓ and fixed signal variance $\sigma_f^2 = 1$. The dashed line indicates the zero prior mean.

The standard SE kernel is isotropic, meaning it treats all input dimensions equally. However, with **Automatic Relevance Determination (ARD)**, each input dimension d is assigned its own length-scale (i.e. $\ell = (\ell_1,\, \ell_2,\, \ell_3 \,....,\, \ell_d))$, resulting in an anisotropic kernel

$$k_{\mathrm{SE,ARD}}(\mathbf{x}, \mathbf{x}') = \sigma_f^2 \exp\left(- \sum_{d=1}^{D} \frac{(x_d - x'_d)^2}{2\ell_d^2} \right) \tag{2.5}$$

This ARD variant allows the model to infer the relevance of each input dimension, assigning longer length-scales to less relevant dimensions that result in little variation along those axes, and shorter length-scales to more relevant dimensions that allowing variation in those directions. This is shown in Figure 2.2

The SE kernel is stationary, meaning it depends only on the Euclidean distance between points, ensuring translation invariance.

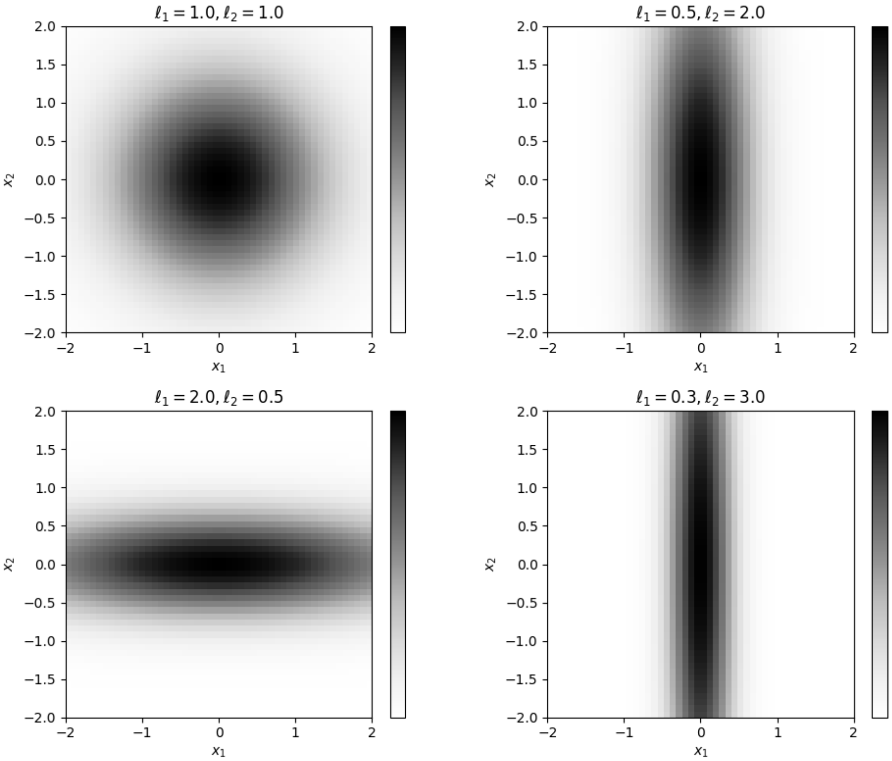


Figure 2.2: Grayscale visualization of the Squared Exponential (SE) kernel with Automatic Relevance Determination (ARD). Each subplot illustrates kernel values $k_{\mathrm{SE,ARD}}(\mathbf{x}, \mathbf{x}')$ for $\mathbf{x}' = (0, 0)$ under different dimension-specific length-scales. Shorter ℓ_d indicates higher sensitivity along that input dimension.

Matérn Kernel

The Matérn kernel family introduces flexibility in smoothness assumptions through an additional parameter [4]. Its general form for a distance $r = |x - x'|$ is

$$k_{\mathrm{Matérn}}(r) = \sigma_f^2 \frac{2^{1-\nu}}{\Gamma(\nu)} \left(\sqrt{2\nu}\, \frac{r}{\ell} \right)^{\nu} K_\nu\!\left(\sqrt{2\nu}\, \frac{r}{\ell} \right) \tag{2.6}$$

where $\nu > 0$ is a shape parameter, $\Gamma(\cdot)$ is the Gamma function, and $K_\nu(\cdot)$ is a modified Bessel function of the second kind.

Like the RBF, ℓ is a length-scale and σ_f^2 a variance. The Matérn class includes the RBF as a limiting case – as $\nu \to \infty$, the Matérn kernel converges to the squared exponential form.

Importantly, ν controls the smoothness of the functions in the GP prior. A GP with a Matérn covariance is $(\lfloor \nu \rfloor)$-times mean-square differentiable (i.e. roughly $\nu - 1$ times differentiable). In simpler terms, larger ν yields smoother functions.

Common choices are half-integer values of ν, for which the kernel has closed-form expressions:

- $\nu = \frac{1}{2}$: The Ornstein–Uhlenbeck (OU) covariance

$$k(r) = \sigma_f^2 \exp(-\frac{r}{\ell}) \tag{2.7}$$

  Known as the exponential kernel. It produces very rough, non-differentiable sample paths that is continuous but not smooth. The OU kernel encodes an assumption that the function is unsmooth. The observations only inform a small neighborhood – points must be close to be strongly correlated

- $\nu = \frac{3}{2}$: Matérn-3/2 kernel

$$k_{\nu=\frac{3}{2}}(r) = \sigma_f^2 \left(1 + \sqrt{3}\,\frac{r}{\ell}\right) \exp\left(-\sqrt{3}\,\frac{r}{\ell}\right) \tag{2.8}$$

  which yields sample paths that are once differentiable. This kernel is less smooth than RBF but smoother than OU, assuming moderate continuity in the function.

- $\nu = \frac{5}{2}$: Matérn-5/2 kernel

$$k_{\nu=\frac{5}{2}}(r) = \sigma_f^2 \left(1 + \sqrt{5}\,\frac{r}{\ell} + \frac{5\,r^2}{3\,\ell^2}\right) \exp\left(-\sqrt{5}\,\frac{r}{\ell}\right). \tag{2.9}$$

  which yields twice differentiable sample paths. This is smoother than $\nu = 3/2$ but still not as excessively smooth as RBF.

As shown in Figure 2.3, increasing the parameter ν in the Matérn kernel leads to smoother functions. Matérn-$\frac{1}{2}$ produces highly non-smooth paths, while Matérn-$\frac{3}{2}$ and $\frac{5}{2}$ yield increasingly smooth trajectories. These kernels offer a flexible alternative to the RBF kernel, imposing less strict smoothness assumptions and better capturing functions with local variations.

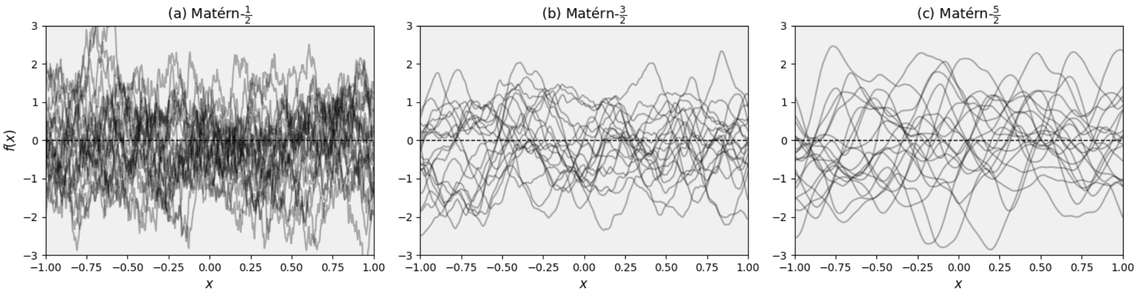


Figure 2.3: Samples from GP priors using Matérn kernels with different smoothness levels ν. From left to right: (a) Matérn-$\frac{1}{2}$, (b) Matérn-$\frac{3}{2}$, and (c) Matérn-$\frac{5}{2}$. As ν increases, the functions become more smoother.

Periodic Kernel

The periodic kernel is designed to model functions that repeat with some fixed period [6]. One common form of a strictly periodic kernel is derived by mapping inputs through sine/cosine features. For input points $\mathbf{x}$ and $\mathbf{x}'$, a typical periodic kernel is

$$k_{\text{Per}}(\mathbf{x}, \mathbf{x}') = \sigma_f^2 \exp\left(-\frac{2\, \sin^2\left(\frac{\pi\, |\mathbf{x} - \mathbf{x}'|}{p}\right)}{\ell^2} \right). \tag{2.10}$$

Here p is the period of repetition, and ℓ is a length-scale parameter controlling how fast the covariance decays with misalignment in phase. Like other kernels, σ_f^2 scales the variance.

It defines a covariance that depends on $\sin^2(\frac{\pi|\mathbf{x}-\mathbf{x}'|}{p})$, which is 0 when $\mathbf{x}$ and $\mathbf{x}'$ differ by an integer multiple of the period p. Thus $k_{\text{Per}}(\mathbf{x}, \mathbf{x}') = \sigma_f^2$ (maximum covariance) whenever $\mathbf{x}' = \mathbf{x} + \mathbf{mp}$ for some integer m, enforcing perfect periodicity of the GP prior with period p. The kernel is stationary on a wrapped domain (the circle of circumference p). Figure 2.4 illustrates sample functions drawn from a GP prior with the periodic kernel, showing the repeating structure induced by the kernel.

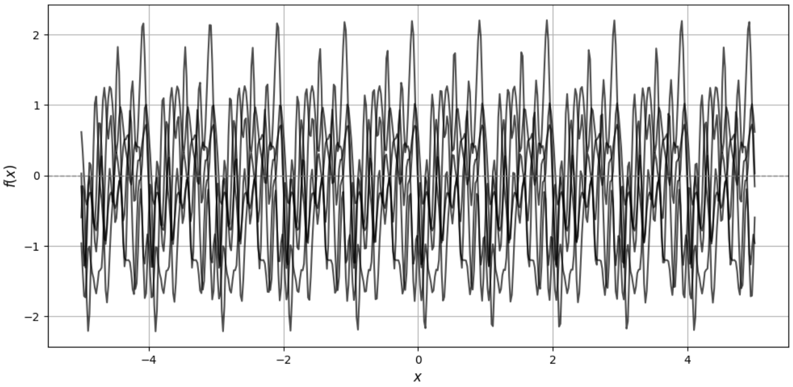


Figure 2.4: Samples from a GP prior using a periodic kernel with period $p = 1$ and length-scale $\ell = 0.3$. The periodic kernel enforces strict periodicity, with maximum covariance when inputs differ by integer multiples of the period. Shorter length-scales result in sharper alignment around each phase, as seen in the repeating structure of the sample paths.

2.1.2 Gaussian Process Posterior

A fundamental aspect of GPs is their inherent connection to Bayesian inference—the ability to make predictions and update beliefs as new information becomes available. In the GP posterior, new observations are incorporated to refine the model. In this section, we discuss the process of updating the GP posterior in both noise-free and noise-added observation scenarios.

Posterior Distribution Derivation

In GP regression, we consider a noise-free training dataset $D = (\mathbf{x}^{(i)}, f^{(i)})i = 1^n$ and test points $\mathbf{X}_*$, where the goal is to predict the function values f_* at $\mathbf{X}_*$. Under a zero-mean GP prior, the joint prior distribution of training and test outputs follows:

$$\begin{bmatrix} y \\ f_* \end{bmatrix} \sim \mathcal{N}\left(0, \begin{bmatrix} k(\mathbf{X},\mathbf{X}) & k(\mathbf{X},\mathbf{X}_*) \\ k(\mathbf{X}_*,\mathbf{X}) & k(\mathbf{X}_*,\mathbf{X}_*) \end{bmatrix}\right). \tag{2.11}$$

Here, $k(\mathbf{X},\mathbf{X})$ represents the covariance matrix for training inputs, $k(\mathbf{X}_*,\mathbf{X}_*)$ for test inputs, and $K(\mathbf{X},\mathbf{X}_*)$ the cross-covariance between them. The kernel function k defines prior assumptions on smoothness and continuity.

Applying Gaussian conditioning, the posterior mean and covariance functions become:

$$\mu_D(X_*) = k(\mathbf{X}_*,\mathbf{X})k(\mathbf{X},\mathbf{X})^{-1}f \tag{2.12}$$

$$k_D(\mathbf{X}_*,\mathbf{X}_*) = k(\mathbf{X}_*,\mathbf{X}_*) - k(\mathbf{X}_*,\mathbf{X})k(\mathbf{X},\mathbf{X})^{-1}k(\mathbf{X},\mathbf{X}_*) \tag{2.13}$$

The **posterior mean** $\mu_D(X_*)$ represents our best estimate of the function's value at X_* given the observed data. It is a weighted linear combination of the training observations f, where the weights are determined by the covariance between X_* and the training inputs. This means the model uses nearby known data points to make predictions, assigning higher importance to observations that are more strongly correlated with X_*.

The **posterior covariance** $k_D(\mathbf{X}_*,\mathbf{X}_*)$ describes the uncertainty in the predictions.

This means that the model uncertainty is significantly reduced near observed data points, where it can rely on the support of known values to make confident predictions. In contrast, as we move further away from the observed points, the uncertainty in the predictions increases. This rising uncertainty reflects the model's limited information about these regions, indicating less confidence in its predictions due to the lack of direct data support, as shown in Figure 2.5.

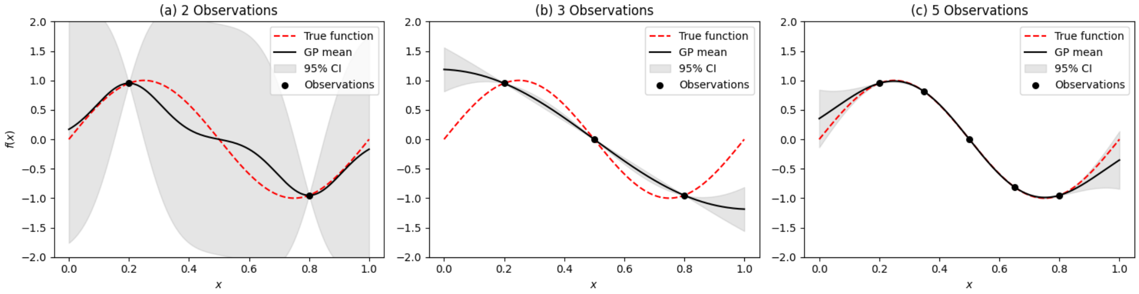


Figure 2.5: Illustration of GP posterior updates as more observations are added. The true function (red dashed) is $f(x) = \sin(2\pi x)$. The GP mean (black line) and 95% confidence interval (gray band) adjust with new data points. As the number of observations increases, the GP becomes more confident and accurate in regions close to the data.

In practical scenarios, observations are often contaminated with noise. This is modeled as:

$$y = f(\mathbf{x}) + \varepsilon, \quad \varepsilon \sim \mathcal{N}(0, \nu^2), \tag{2.14}$$

where ε represents Gaussian noise with variance ν^2. As a result, the covariance of the observed outputs becomes:

$$\operatorname{cov}(y \mid \mathbf{X}) = k(\mathbf{X}, \mathbf{X}) + \nu^2 I, \tag{2.15}$$

where the term $\nu^2 I$ accounts for independent, identically distributed noise in each observation. Using this updated covariance structure, the posterior predictive distribution is also modified. The posterior mean function is given by:

$$\mu_D(\mathbf{X}_*) = k(\mathbf{X}_*, \mathbf{X}) \left[k(\mathbf{X}, \mathbf{X}) + \nu^2 I\right]^{-1} y, \tag{2.16}$$

and the posterior covariance function becomes:

$$k_D(\mathbf{X}_*, \mathbf{X}_*) = k(\mathbf{X}_*, \mathbf{X}_*) - k(\mathbf{X}_*, \mathbf{X}) \left[k(\mathbf{X}, \mathbf{X}) + \nu^2 I\right]^{-1} k(\mathbf{X}, \mathbf{X}_*). \tag{2.17}$$

In the presence of noise, the model no longer strictly interpolates the training data. Instead, it produces smoother predictions that account for the uncertainty introduced by the noise.

To determine suitable hyperparameters—such as the kernel's lengthscale and variance—it is common to maximize the log-marginal likelihood of the observed data:

$$\log p(y \mid \mathbf{X}) = -\frac{1}{2} y^\top (K + \sigma_n^2 I)^{-1} y - \frac{1}{2} \log \left|K + \sigma_n^2 I\right| - \frac{n}{2} \log 2\pi. \tag{2.18}$$

where $K = k(\mathbf{X}, \mathbf{X})$ is the kernel (covariance) matrix, and σ_n^2 is the noise variance.

This objective function strikes a balance between model fit and complexity. However, it is generally non-convex, so multiple optimization restarts are often employed to avoid getting trapped in poor local minima.

2.2 Bayesian Optimization

BO is a strategy for global optimization of expensive, black-box functions that lack closed-form expressions or easy derivatives [5] [7]. It addresses problems where each function evaluation is costly (e.g., time-consuming experiments or expensive simulations), so the goal is to find the global optimum in as few evaluations as possible.

BO approaches this by building a probabilistic surrogate model of the objective function and using it to decide where to sample next. In essence, BO consists of two key components: (1) a Bayesian statistical model (usually a GP) that captures our current belief about the unknown function which had already introduced in the previous subsection, and (2) an acquisition function that uses this model to determine the next query point. This framework enables a principled exploration-exploitation trade-off – exploring uncertain regions while exploiting areas predicted to have high objective values.

One example can be shown in Figure 2.6 that illustrates how the sample acquisition function, Expected Improvement, guides the search in BO. In early iterations, EI favors exploration in uncertain regions. As more points are observed, the GP model refines its estimate, and EI focuses around the likely minimum.

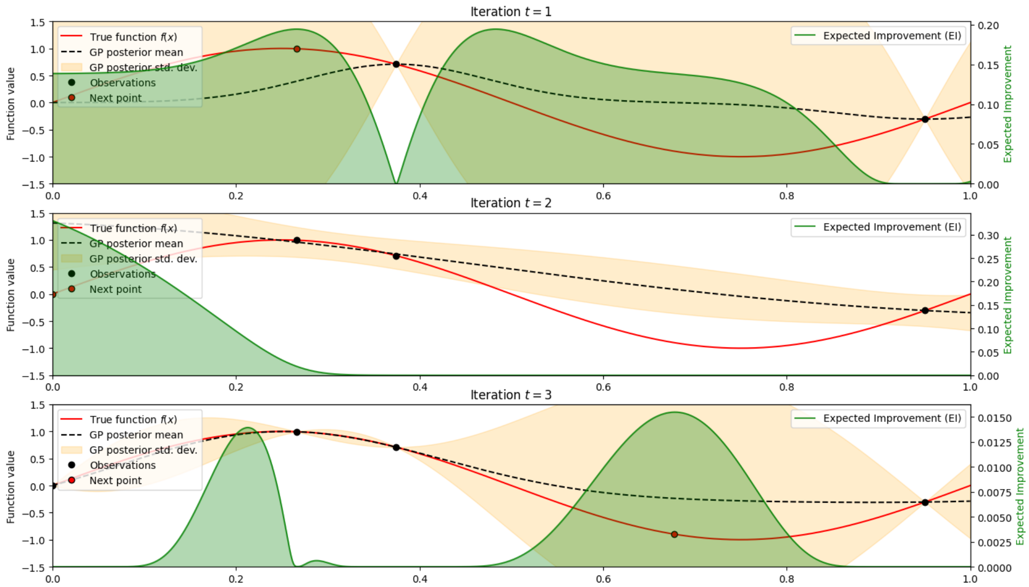


Figure 2.6: Visualization of three iterations of BO using the Expected Improvement (EI) acquisition function. The true function $f(\mathbf{x})$ (red) is unknown to the optimizer. At each step, a GP posterior (black dashed line with green uncertainty band) is updated based on observed data (black dots). The EI acquisition function (light green curve) guides the selection of the next evaluation point (orange dot). As the process continues, the GP becomes more confident near the optimum and EI shrinks in regions with low uncertainty or low expected gain.

2.2.1 Improvement Based Acquisition Functions

With a GP posterior in place, BO needs a policy to decide the next evaluation point. This role is played by *acquisition functions* (also known as *infill criteria* or *utility functions*).

An acquisition function $a(\mathbf{x})$ maps the GP's predictive distribution at x to a real value that reflects the utility of sampling at x. It is designed such that maximizing $a(\mathbf{x})$ (approximately) corresponds to selecting points that are expected to improve our objective. Crucially, acquisition functions depend only on the surrogate (GP) and not on the unknown true function—thus they are cheap to evaluate and can be optimized by standard methods.

Various acquisition functions have been proposed, each with different theoretical motivations. We focus here on three improvement-based choices: *Probability of Improvement (PI)*, *Expected Improvement (EI)*, and *Gaussian Process Upper Confidence Bound (GP-UCB)*.

Probability of Improvement (PI)

The Probability of Improvement (PI) is one of the earliest BO acquisition functions [8]. PI chooses the next evaluation point by maximizing the probability that a new sample will outperform the current best observed value.

Let $f_n^* = \max\{y_1, \ldots, y_n\}$ be the best (maximum) outcome observed so far. The PI acquisition function is defined as:

$$a_{\mathrm{PI}}(\mathbf{x}) = \Pr\left(f(\mathbf{x}) > f_n^* + \xi \mid \mathcal{D}_n\right) \tag{2.19}$$

where $\xi \geq 0$ is a slack (or exploration) parameter that can encourage exploration.

Assuming a GP posterior, $f(\mathbf{x}) \sim \mathcal{N}(\mu_n(x), \sigma_n^2(x))$, the probability can be evaluated in closed form as a Gaussian tail probability:

$$a_{\mathrm{PI}}(\mathbf{x}) = \Phi\left(\frac{\mu_n(\mathbf{x}) - (f_n^* + \xi)}{\sigma_n(\mathbf{x})}\right) \tag{2.20}$$

where $\Phi(\cdot)$ denotes the standard normal cumulative distribution function.

The acquisition function $a_{\mathrm{PI}}(x)$ tends to be high at points where the predicted mean significantly exceeds the current best value f_n^*, indicating a high likelihood of improvement. It can also be high in regions with large predictive uncertainty, which increases the chance of surpassing f_n^*.

PI only considers the probability of achieving any improvement, without accounting for the magnitude of the improvement. In other words, all improvements are treated equally, regardless of how substantial they are. PI may overly concentrate on regions that have a relatively high chance of minor improvement while ignoring areas with the potential for large improvements but lower probability. Consequently, PI often exhibits overly exploitative behavior, which may lead it to become stuck in local optima and fail to explore the search space adequately.

The exploration parameter ξ (sometimes referred to as "kappa" in the literature) can partially address this issue. Increasing ξ raises the threshold for improvement, thereby encouraging exploration. When $\xi = 0$, PI reduces to maximizing the probability $\Pr(f(\mathbf{x}) > f_n^*)$.

Expected Improvement (EI)

Expected Improvement is a refined acquisition function that overcomes the main limitation of PI by accounting for the magnitude of improvement. EI is defined as the expected gain in the objective value if sampling at x, relative to the current best f_n^* [9] [10].

Formally, using the same $f_n^* = \max\{y_i \mid i \leq n\}$ for a maximization problem, the improvement if we sample at x is

$$(f(\mathbf{x}) - f_n^*)_+ = \max\{0, f(\mathbf{x}) - f_n^*\} \tag{2.21}$$

i.e., the amount by which $f(x)$ exceeds the incumbent best (or 0 if it does not). The expected improvement is then the posterior expectation of this quantity:

$$a_{\mathrm{EI}}(\mathbf{x}) = \mathbb{E}_n\left[(f(\mathbf{x}) - f_n^*)_+\right] \tag{2.22}$$

where $\mathbb{E}_n[\cdot]$ denotes expectation under the GP posterior $f(\mathbf{x}) \sim \mathcal{N}(\mu_n(\mathbf{x}), \sigma_n^2(\mathbf{x}))$.

By taking the expectation, EI smoothly trades off the probability of improvement with the size of improvement. Points that have a small chance of greatly exceeding f_n^* can thus score as well as points that will most likely yield a small improvement. One of the strengths of EI is that it has a closed-form expression for Gaussian posteriors. Using analytical integration by parts, one obtains:

$$a_{\mathrm{EI}}(\mathbf{x}) = (\mu_n(\mathbf{x}) - f_n^*)\Phi(Z) + \sigma_n(\mathbf{x})\phi(Z) \tag{2.23}$$

where $Z := \frac{\mu_n(\mathbf{x}) - f_n^*}{\sigma_n(\mathbf{x})}$, and $\phi(\cdot)$ and $\Phi(\cdot)$ are the standard normal PDF and CDF, respectively.

The first term, $(\mu_n(\mathbf{x}) - f_n^*)\Phi(Z)$, increases with a larger predicted mean $\mu_n(\mathbf{x})$ and reflects the exploitation component of the acquisition function, favoring points expected to outperform the current best.

The second term, $\sigma_n(\mathbf{x})\phi(Z)$, grows with greater predictive uncertainty $\sigma_n(\mathbf{x})$ and captures the exploration component, encouraging sampling in less well-understood regions.

Together, these terms ensure that EI naturally balances the trade-off between exploitation and exploration. If $\mu_n(\mathbf{x})$ is well below f_n^*, Z is negative and $\Phi(Z)$ is small, but the second term $\sigma_n(\mathbf{x})\phi(Z)$ (approximately proportional to $\sigma_n(\mathbf{x})$ when Z is small) ensures points with high uncertainty can still yield a decent EI. Conversely, if $\mu_n(\mathbf{x})$ is much larger than f_n^*, then even with small $\sigma_n(\mathbf{x})$, Z is large and $\Phi(Z) \approx 1$, so $a_{\mathrm{EI}}(x) \approx \mu_n(\mathbf{x}) - f_n^*$—close to the predicted improvement.

This behavior encourages sampling in regions that are either promising in value or poorly understood, quantitatively embodying the exploration–exploitation trade-off. In fact, curves of constant EI in the (μ, σ) plane show an implicit trade-off: a point with moderate mean and high variance can equal the EI of a point with higher mean but lower variance.

Gaussian Process Upper Confidence Bound (GP-UCB)

The Upper Confidence Bound (UCB) acquisition function comes from a different perspective, the multi-armed bandit theory, and provides strong theoretical guarantees. GP-UCB is based on the principle of optimism in the face of uncertainty.

The idea is to select the point that maximizes an upper confidence bound on the objective's value. Specifically, given the GP posterior $\mathcal{N}(\mu_n(\mathbf{x}), \sigma_n^2(\mathbf{x}))$, an upper $(1-\delta)$-confidence bound for the unknown $f(\mathbf{x})$ can be written as $\mu_n(\mathbf{x}) + \sqrt{\beta_n}\sigma_n(\mathbf{x})$, where β_n is a multiplier (dependent on δ and past data) that determines the confidence level. The GP-UCB policy chooses the next point as:

$$\mathbf{x}_{n+1} = \arg\max_{x} \left\{\mu_n(\mathbf{x}) + \sqrt{\beta_n}\sigma_n(\mathbf{x})\right\} \tag{2.24}$$

In other words, at each step it picks the location where the sum of the predictive mean and a scaled uncertainty is highest. The intuition is clear: $\mu_n(\mathbf{x})$ promotes points with high predicted reward (exploitation), while the $\sqrt{\beta_n}\sigma_n(\mathbf{x})$ term incentivizes exploration of uncertain points.

By adjusting β_n over time (typically a sequence growing at a rate related to $\ln n$), one can ensure that eventually the true optimum will be sampled with high probability. In fact, its proved that under an appropriate choice of β_n, the GP-UCB algorithm enjoys a sub-linear regret bound [11].

GP-UCB has solid theoretical underpinnings: under certain smoothness conditions on the objective (captured by the GP prior), the sequence of points chosen by GP-UCB will converge to the global maximizer of $f(\mathbf{x})$ with high probability. This makes GP-UCB particularly attractive in scenarios where theoretical assurance is important (e.g., critical engineering systems or when designing algorithms with performance guarantees). In practice, a simplified version of GP-UCB often uses a constant β (treated as a tunable parameter) and works well even without an increasing schedule, although the strict regret guarantees apply to the properly scheduled version.

It is worth mentioning that UCB is typically described in terms of maximizing f; for minimization problems, one would use a *Lower Confidence Bound* (LCB), defined as $\mu_n(\mathbf{x}) - \sqrt{\beta_n}\sigma_n(\mathbf{x})$. The theoretical property of eventual convergence holds analogously for the minimization case.

Chapter 3

Entropy Based Search Policies

3.1 Introduction

Information-theoretic acquisition functions take a different approach: rather than directly optimizing an expected objective improvement or an upper bound, they aim to reduce uncertainty about the location of the global optimum. Entropy Search (ES) and Predictive Entropy Search (PES), Max Entropy Search(MES) are three such methods. They choose the next sample to maximally gain information about the argmax of $f(\mathbf{x})$, as quantified by entropy. These methods are more complex and computationally intense, but conceptually powerful.

This chapter intends to investigate the performance of the entropy-based acquisition functions compared to improvement based methods.

3.2 Background

3.2.1 Information Gain In Bayesian Optimization

Entropy is a fundamental concept from information theory that quantifies uncertainty in a probability distribution. Entropy $H[p]$ is defined as the negative expected log-probability of outcomes [12]:

$$H[p] = -\sum_i p_i \log p_i \tag{3.1}$$

Intuitively, entropy measures the "spread" or unpredictability of a distribution—a high entropy value indicates great uncertainty. For continuous distributions, a well-defined analog is the relative entropy or *Kullback-Leibler (KL) divergence* with respect to a reference measure [13]

$$D_{\mathrm{KL}}(P\|Q) = \int_{-\infty}^{\infty} p(x) \log\left(\frac{p(\mathbf{x})}{q(\mathbf{x})}\right) dx \tag{3.2}$$

Entropy's interpretation as a measure of information content makes it a natural choice for guiding learning algorithms: reductions in entropy correspond to gains in information. In Bayesian terms, the expected information gain from an experiment can be quantified by the drop in entropy of the posterior distribution relative to the prior.

3.2.2 Approximation Technique

Entropy-based search methods in BO aim to select query points that maximize the expected information gain about the global optimum. However, calculating this information gain often involves complex integrals over high-dimensional distributions, which are analytically intractable.

This is where approximation techniques become essential—they make the computation feasible by approximating difficult terms like the posterior over the global optimum or the expected reduction in entropy.

Monte Carlo Estimation

Monte Carlo (MC) methods are widely used to approximate the expectations defining the acquisition. The idea is to sample from the GP posterior to approximate the distribution of the optimum [14].

The mutual information is then estimated by averaging entropy computations over these samples. The accuracy of MC integration improves with more samples at the cost of runtime. A trade-off arises: finer accuracy vs. computational cost.

Typically a few thousand function samples may be needed for stable estimates in high dimensions, which is expensive but still far cheaper than a brute-force integration over a continuous domain. Techniques like importance sampling can sometimes improve efficiency by focusing samples in informative regions.

Expectation Propagation

Expectation Propagation (EP) is a deterministic approximation algorithm designed to estimate posterior distributions in probabilistic models where exact inference is computationally intractable [15]. Unlike sampling-based methods such as Markov Chain Monte Carlo (MCMC), EP iteratively constructs a tractable approximation to the true posterior by minimizing the Kullback–Leibler (KL) divergence in a local and moment-matching manner.

The central idea behind EP is to approximate each non-conjugate factor in the posterior with a simpler, tractable distribution, usually chosen from the exponential family such as Gaussians. The result is an efficient inference scheme that maintains computational tractability while offering accurate approximations to posterior moments, particularly in models like GPs where analytic treatment of the prior is feasible but likelihoods may be non-Gaussian.

Given a probabilistic model with latent variables $\mathbf{z}$ and a set of observations $\mathcal{D} = \{x_i, y_i\}_{i=1}^N$, the true posterior is expressed as

$$p(\mathbf{z} \mid \mathcal{D}) = \frac{1}{Z} p(\mathbf{z}) \prod_{i=1}^{N} p(y_i \mid z_i) \tag{3.3}$$

where $p(\mathbf{z})$ is the prior and $p(y_i \mid z_i)$ are the likelihood terms. EP approximates this posterior using a simpler distribution

$$q(\mathbf{z}) \propto p(\mathbf{z}) \prod_{i=1}^{N} \tilde{t}_i(z_i) \tag{3.4}$$

where each $\tilde{t}_i(z_i)$ is a Gaussian site function approximating the original likelihood term. This ensures that the product $q(\mathbf{z})$ remains Gaussian and tractable, enabling efficient computation of marginals and moments.

The EP algorithm proceeds by iteratively updating each site factor. For a given site i, EP temporarily removes the site from the approximate posterior to form a *cavity distribution*

$$q^{\setminus i}(z_i) \propto \frac{q(z_i)}{\tilde{t}_i(z_i)} \tag{3.5}$$

which captures the influence of all other factors. This cavity distribution is then multiplied by the original factor $p(y_i \mid z_i)$ to form a *tilted distribution*

$$\hat{p}(z_i) \propto q^{\setminus i}(z_i) \cdot p(y_i \mid z_i) \tag{3.6}$$

Since $\hat{p}(z_i)$ may no longer be Gaussian, EP approximates it with a Gaussian distribution by matching the first and second moments (mean and variance). The updated moments are then used to compute new site parameters such that when recombined with the cavity distribution, the resulting marginal matches the tilted distribution in terms of these moments.

EP is well-suited for models like Gaussian Process classification, where the prior $p(\mathbf{f})$ is Gaussian but the likelihood—such as the probit likelihood $\Phi(y_i f_i)$—is not. In such cases, EP approximates each likelihood term with a Gaussian, enabling closed-form posterior inference. This is crucial for applications like BO, where one needs accurate estimates of the marginal predictive distribution $p(y \mid \mathbf{x}, \mathcal{D})$ to compute acquisition functions such as Expected Improvement or entropy-based criteria. EP also plays an important role in entropy-based BO strategies such as Entropy Search and Predictive Entropy Search, where it enables tractable approximations of mutual information or expected information gain under GP priors.

Gumbel Sampling

Gumbel-based sampling which provides a tractable approximation for distribution of global maximum under certain assumptions [16].

Specifically, when considering a finite set of input locations $\{\mathbf{x_1}, \mathbf{x_2}, \mathbf{x_3}, \ldots, \mathbf{x_n}\} \subset \mathcal{X}$, and assuming the posterior predictive distribution of the GP at these locations yields independent function values $f(\mathbf{x_i}) \sim \mathcal{N}(\mu_i, \sigma_i^2)$, the distribution of the maximum y^* over this set can be approximated using extreme value theory.

The key idea is that the maximum of a finite set of i.i.d. or weakly dependent Gaussian variables converges to a *Gumbel distribution* in the limit. For finite n, one can approximate the distribution of y^* as:

$$p(y^*) \approx \text{Gumbel}(\mu, \beta) \tag{3.7}$$

where the location and scale parameters μ and β can be estimated empirically from samples of $\max_i f(x_i)$, or via analytical approximations when available. Given this approximation, one can efficiently generate samples $\{y_j^*\}_{j=1}^M \sim p(y^*)$ without drawing full posterior function realizations.

These samples can then be used to compute mutual information terms of the form:

$$I(f(x); y^*) = H(f(x)) - \mathbb{E}_{y^*}[H(f(x) \mid y^*)] \tag{3.8}$$

where $H(f(x))$ is the entropy of the GP marginal at x, and the conditional entropy $H(f(x) \mid y^*)$ is given by the entropy of a Gaussian truncated above y^*:

$$H(f(\mathbf{x}) \mid y^*) = \frac{1}{2}\log(2\pi e\sigma^2(\mathbf{x})) - \frac{\phi(z)}{2\Phi(z)} \cdot z \tag{3.9}$$

with $z = \frac{y^*-\mu(\mathbf{x})}{\sigma(\mathbf{x})}$, and ϕ, Φ denoting the standard normal PDF and CDF, respectively. Averaging over the y_j^* samples gives an efficient and scalable entropy approximation.

Compared to standard Monte Carlo integration over function samples, this method significantly reduces computational cost while still capturing key aspects of the distribution of the optimum. However, the independence assumption may degrade the quality of approximation when posterior correlations are non-negligible.

3.3 Entropy-based search methods

3.3.1 Entropy Search (ES)

Entropy Search (ES) is a landmark algorithm that embodies an information-theoretic approach to BO [17]. ES begins by defining a probability distribution $p(\mathbf{x}^* \mid D_n)$ over the location of the global optimum x^* (typically the minimizer or maximizer of an unknown function), given the current data D_n. This distribution captures the algorithm's uncertainty about the location of the true optimum.

The acquisition function in ES is defined as the expected reduction in the entropy of this distribution after a hypothetical evaluation at a candidate point $\mathbf{x}$. Formally, it is expressed as:

$$\alpha_{\text{ES}}(\mathbf{x}) = H[p(\mathbf{x}^* \mid D_n)] - \mathbb{E}_{y\sim p(y|D_n,x)}\left[H[p(\mathbf{x}^* \mid D_n \cup \{(\mathbf{x}, y)\})]\right] \tag{3.10}$$

where $H[\cdot]$ denotes the entropy. The first term $H[p(\mathbf{x}^* \mid D_n)]$ represents the current uncertainty about the optimizer and is constant with respect to $\mathbf{x}$. The second term is the expected residual uncertainty after observing a new function value $y = f(\mathbf{x})$, averaged over the predictive distribution of y at $\mathbf{x}$. By maximizing $\alpha_{\text{ES}}(x)$, the algorithm selects the evaluation point expected to yield the greatest information gain about the location of the global optimum.

Equivalently, ES seeks to maximize the *mutual information* between the new observation and the unknown optimum location $\mathbf{x}^*$, a perspective that was emphasized in subsequent work.

In practice, computing $p(\mathbf{x}^* \mid D_n)$ exactly is intractable because it involves reasoning about the argmax of a random function. ES addresses this by modeling the unknown function with a GP surrogate and approximating the optimizer distribution using methods such as Monte Carlo simulation or Expectation Propagation (EP).

To make the optimization computationally feasible, ES often incorporates a pre-filtering step using simpler acquisition functions like *Probability of Improvement (PI)* to narrow down the candidate set before applying the full ES acquisition calculation. This helps manage computational cost, especially in high-dimensional or expensive-to-evaluate problems.

Entropy is measured in a *relative* sense using the negative Kullback-Leibler (KL) divergence from a uniform reference distribution over the domain. This means that a uniform belief (i.e., complete uncertainty) incurs the highest loss, and as the optimizer distribution becomes more concentrated (i.e., entropy decreases), the loss diminishes. ES thus selects query points expected to maximally reduce this loss, or equivalently, to maximally increase information about x^*.

3.3.2 Predictive Entropy Search (PES)

Predictive Entropy Search (PES) builds upon the same core idea as Entropy Search (ES), but introduces a crucial theoretical reframing to make the acquisition function significantly more tractable to compute [18].

The key insight in Predictive Entropy Search (PES) is to exploit the symmetry of mutual information: the mutual information between the location of the global minimizer $\mathbf{x}^*$ and a new observation y can be expressed in two equivalent ways. One formulation, as in Entropy Search (ES), defines it as the expected reduction in the entropy of the posterior distribution over the minimizer $p(\mathbf{x}^* \mid D_n)$ after observing y. The alternative, used in PES, interprets it as the expected reduction in the entropy of the predictive distribution $p(y \mid D_n, \mathbf{x})$ after conditioning on knowledge of $\mathbf{x}^*$. These perspectives yield the same mutual information but differ in computational tractability and implementation.

The ES acquisition function $\alpha_{\mathrm{ES}}(\mathbf{x})$ can be rewritten in a mathematically equivalent but computationally simplified "predictive" form through algebraic manipulation:

$$\alpha_{\mathrm{PES}}(\mathbf{x}) = H[p(y \mid D_n, \mathbf{x})] - \mathbb{E}_{\mathbf{x}^* \sim p(\mathbf{x}^* \mid D_n)} \left[H[p(y \mid D_n, \mathbf{x}, \mathbf{x}^*)] \right] \tag{3.11}$$

where $H[\cdot]$ denotes entropy, and the expectation is now taken over samples of $\mathbf{x}^*$ from the current belief. This formulation swaps the integration order compared to ES, avoiding the need to recompute the posterior over $\mathbf{x}^*$ for every possible observation y. Instead, PES samples plausible optima $\mathbf{x}^* \sim p(\mathbf{x}^* \mid D_n)$ and computes the information gain that a query at $\mathbf{x}$ would provide under each hypothesized true optimum.

This reframing makes PES much more computationally efficient. In practice, PES approximates $p(\mathbf{x}^* \mid D_n)$ by drawing samples from the posterior GP, often via Monte Carlo simulation (e.g., drawing GP function realizations and computing their argmax). To maintain tractability, PES also supports other approximation methods such as Expectation Propagation (EP) for certain conditional entropy computations.

As with ES, computing the PES acquisition function can be expensive in high-dimensional or large-data settings. To mitigate this, PES implementations often include a **pre-filtering step** using simpler utility functions like *Expected Improvement (EI)* or *Probability of Improvement (PI)* to select a small set of promising candidate points for full evaluation under the PES criterion.

PES retains the key advantage of ES, explicitly maximizing the mutual information between the next query and the global optimum, but it does so in a more computationally efficient manner by focusing on the entropy of the predictive observation space rather than the belief over the optimizer.

3.3.3 Max Entropy Search (MES)

Max-value Entropy Search (MES) is an information-theoretic acquisition strategy that selects query points by maximizing the mutual information between a potential observation and the unknown optimum value $y^* = \max_x f(\mathbf{x})$. By shifting the focus from the location of the optimum $\mathbf{x}^*$ to the scalar output y^*, MES avoids high-dimensional entropy estimation and significantly improves computational tractability.

Formally, MES selects the next evaluation point x by maximizing:

$$\alpha_{\mathrm{MES}}(\mathbf{x}) = I(f(\mathbf{x}); y^* \mid \mathcal{D}_n),$$

where $\mathcal{D}_n$ denotes the current set of observations. This mutual information can be expressed as a difference in entropies:

$$\alpha_{\mathrm{MES}}(\mathbf{x}) = \mathcal{H}(f(\mathbf{x}) \mid \mathcal{D}_n) - \mathbb{E}_{y^*}[\mathcal{H}(f(\mathbf{x}) \mid y^*, \mathcal{D}_n)],$$

where $\mathcal{H}$ denotes differential entropy. The second term corresponds to the expected entropy of $f(\mathbf{x})$ conditioned on a given value of the maximum, effectively treating y^* as a threshold.

To approximate the distribution of y^*, MES employs Gumbel-based sampling methods over the posterior of the GP surrogate. For further efficiency, candidate points may be pre-filtered using lightweight heuristics before computing the acquisition value.

MES is particularly well-suited for high-dimensional settings due to its reduced reliance on input-space sampling. It has also been shown to admit theoretical regret bounds, marking a significant development in the formal understanding of information-theoretic optimization methods.

Chapter 4

Numerical Studies

To systematically evaluate the performance of various acquisition strategies in BO, we developed a controlled experimental framework using PyTorch and GPyTorch. The experiment compared six acquisition functions: LCB, EI, PI, ES, PES, and MES.

For each acquisition function, we assessed performance across varying problem dimensionalities (5, 10, and 20 dimensions) and tested them against several standard benchmark objective functions, including Rosenbrock, Ackley, and Schwefel. This setup allows us to analyze how each acquisition strategy scales with complexity and generalizes across different optimization landscapes.

4.1 Experiment Setup

A complete BO framework was implemented to support systematic experimentation under controlled conditions. For each experimental run, the optimization process begins with an initial training dataset generated by sampling 20 points randomly from a bounded domain of [-1, 1]. Gaussian noise $\sigma^2 = 0.01$ is added to simulate the observation noises.

The BO framework trains a GP model as the surrogate for the unknown objective function for each round of iteration. The GP model is trained via exact marginal log likelihood, using the LBFGS optimizer across 10 iterations to ensure convergence. In each BO iteration, the acquisition function is optimized to identify the next evaluation point, which is then queried using the true objective function. The resulting observation is appended to the training set, and the GP model is subsequently retrained to incorporate the new information.

This iterative process is repeated for 200 iterations, during which relevant metrics such as the best observed value, the number of improvements are recorded.

Each acquisition function was evaluated independently over ten experimental runs to account for stochastic variability and ensure statistical robustness. For each run, the initial training set was held fixed across all acquisition functions to ensure a fair comparison. Each acquisition function began its optimization with 100 pre-selected candidate points.

In the constructed acquisition functions for the experiment, the Lower Confidence Bound (LCB) and Probability of Improvement (PI) acquisition functions, the exploration parameter κ was set to 0.01. Both Entropy Search and Predictive Entropy Search utilize PI as an utility function for pre-filtering and Monte Carlo Sampling as the approximation method, while Max Entropy Search

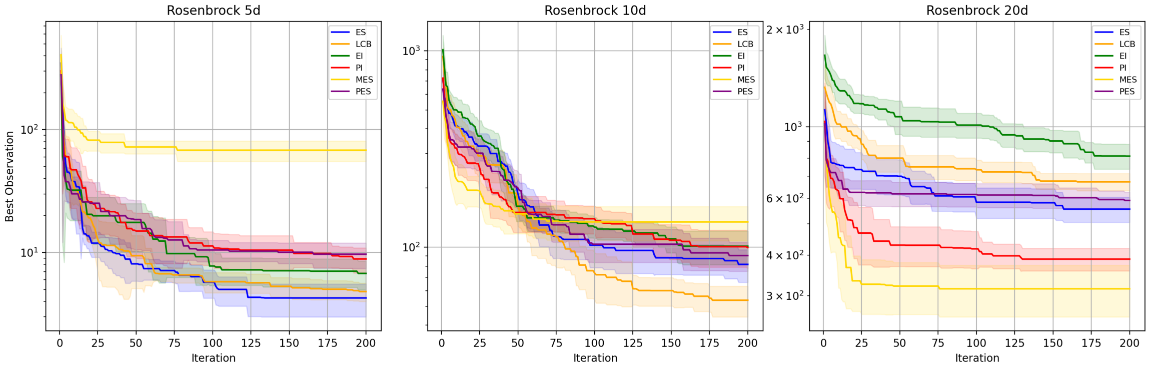


Figure 4.1: Performance comparison of six acquisition functions (ES, LCB, EI, PI, MES, PES) on the Rosenbrock function in 5, 10, and 20 dimensions. The y-axis shows the best function value found so far, averaged over multiple runs, with shaded regions indicating 95% confidence interval

use gumble sampling for approximation.

4.2 Result And Analysis

To evaluate the scalability of acquisition strategies, we assess their performance on the Rosenbrock function across 5, 10, and 20 dimensions. Figure 4.1 presents the best observed values over 200 iterations, averaged across trials, with shaded regions representing 95% confidence interval.

Across all dimensions, PES and MES demonstrate superior performance, particularly in higher dimensions. Notably, both methods exhibit a steep initial decline in objective values, reflecting an accelerated convergence rate during early iterations. This rapid improvement suggests more efficient exploration and exploitation in the initial optimization phase.

In 5D, all methods converge reasonably well; however, PES and MES attain lower objective values more rapidly than EI and LCB, which display more gradual convergence. In 10D and 20D, the performance gap widens. Entropy-based methods maintain steady improvement, while EI and UCB plateau earlier, indicating diminishing effectiveness in higher-dimensional search spaces.

These results highlight the robustness and scalability of entropy-based acquisition functions. Their ability to capture global uncertainty contributes to faster convergence and improved performance as dimensionality increases, making them especially effective for high-dimensional BO.

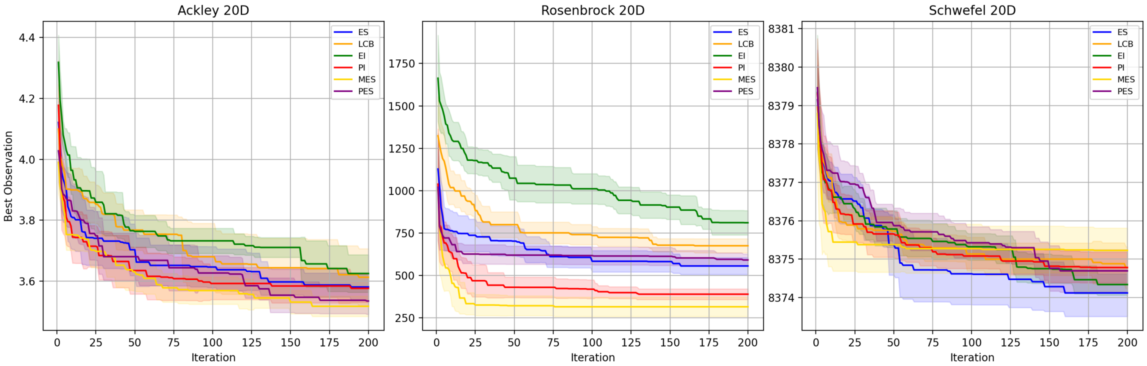
We evaluate acquisition functions on three 20-dimensional benchmark functions: Ackley, Rosenbrock, and Schwefel. Figure 4.2 presents the best observed values over 200 iterations, averaged across trials, with shaded regions representing 95% confidence interval.

Across all cases, PES and MES exhibit consistently strong performance, characterized by a *steep initial decline* and *faster convergence* compared to standard baselines.

On the *Ackley function*, all methods show rapid early improvement, but PES and MES achieve lower final objective values with reduced variance, indicating both efficiency and stability.

The *Rosenbrock function*, known for its narrow curved valley, further highlights the advantage of entropy-based methods. PES achieves the best results, maintaining steady improvement where methods like EI and PI stagnate early.

On the more deceptive *Schwefel function*, convergence is slower overall. Nonetheless, PES and


Ackley 20D

Rosenbrock 20D

Schwefel 20D

Figure 4.2: Performance comparison of six acquisition functions (ES, LCB, EI, PI, MES, PES) on three 20-dimensional benchmark functions: Ackley, Rosenbrock, and Schwefel. Each curve shows the median best function value found so far, with shaded areas representing 95% confidence interval.

MES remain competitive, achieving better final values than EI, PI, and LCB, while ES performs well but with more variability.

These results reinforce the scalability of entropy-based acquisitions. Their ability to model global uncertainty leads to *faster and more reliable convergence*, especially in complex, high-dimensional landscapes.

Chapter 5

Concluding Remarks

5.1 Conclusion

We conducted a systematic evaluation of six acquisition functions, LCB, EI, PI, ES, PES, and MES, within a unified BO framework, focusing on both convergence behavior and scalability. Experiments were performed across multiple benchmark functions (Ackley, Rosenbrock, Schwefel) and dimensionalities (5D, 10D, 20D), using GP surrogates trained with exact marginal log likelihood and acquisition optimization based on pre-filtered candidate sets.

Entropy-based methods, PES and MES, consistently outperformed standard baselines. They exhibited accelerated convergence, especially in the early iterations, and maintained superior performance in higher dimensions. This advantage was most prominent in the Rosenbrock and Schwefel functions, where traditional methods stagnated or displayed high variance.

The improved performance of PES and MES stems from their ability to explicitly model and reduce global uncertainty via information-theoretic criteria. Unlike myopic utility-based acquisitions such as EI or PI, entropy-based strategies promote exploration in informative regions of the search space, leading to more efficient use of evaluation budgets.

These results confirm that entropy-based acquisition functions are highly effective for high-dimensional BO, offering both scalability and robustness in navigating complex landscapes.

5.2 Future Work

An important direction for future work is the integration of hyperparameter optimization into the BO framework. While this study used fixed GP hyperparameters, optimizing them, either via marginal likelihood with advanced priors or fully Bayesian approaches (e.g., MCMC, EP), can improve surrogate accuracy and acquisition effectiveness, especially in noisy or high-dimensional regimes.

Moreover, if time permits, we aim to extend this work to the setting of optimization under uncertainty, where GP models may be employed to represent robustness metrics. In such contexts, entropy-based acquisition functions like PES and MES remain promising, as they inherently account for epistemic uncertainty by quantifying information gain about the global optimum.

Future work may explore the use of these acquisition strategies to guide optimization in the presence of uncertain or robustness-sensitive objectives, potentially improving the reliability and generalizability of BO in practical, high-stakes scenarios.

Bibliography

[1] F. Berkenkamp, A. Krause, and A. P. Schoellig, “Bayesian optimization with safety constraints: safe and automatic parameter tuning in robotics,” *Machine Learning*, vol. 112, no. 10, pp. 3713–3747, 2023. [Online]. Available: https://doi.org/10.1007/s10994-021-06019-1

[2] Y. Zhang, D. W. Apley, and W. Chen, “Bayesian optimization for materials design with mixed quantitative and qualitative variables,” *Scientific Reports*, vol. 10, no. 1, p. 4924, 2020. [Online]. Available: https://doi.org/10.1038/s41598-020-60652-9

[3] E. O. Pyzer-Knapp, “Bayesian optimization for accelerated drug discovery,” *IBM Journal of Research and Development*, vol. 62, no. 6, pp. 2:1–2:7, 2018.

[4] C. E. Rasmussen and C. K. I. Williams, *Gaussian Processes for Machine Learning.* Cambridge, MA: MIT Press, 2006, vol. 1.

[5] R. Garnett, *Bayesian Optimization.* Cambridge: Cambridge University Press, 2023.

[6] D. J. MacKay *et al.*, “Introduction to gaussian processes,” *NATO ASI series F computer and systems sciences*, vol. 168, pp. 133–166, 1998.

[7] P. I. Frazier, “A tutorial on bayesian optimization,” *arXiv preprint arXiv:1807.02811*, 2018, arXiv:1807.02811 [stat.ML]. [Online]. Available: https://doi.org/10.48550/arXiv.1807.02811

[8] H. J. Kushner, “A new method of locating the maximum point of an arbitrary multipeak curve in the presence of noise,” *Journal of Basic Engineering*, vol. 86, no. 1, pp. 97–106, 1964. [Online]. Available: https://doi.org/10.1115/1.3653121

[9] J. Mockus, “On bayesian methods for seeking the extremum,” in *Optimization Techniques IFIP Technical Conference.* Springer, 1975, pp. 400–404.

[10] ——, *Bayesian Approach to Global Optimization: Theory and Applications*, ser. Mathematics and Its Applications. Dordrecht: Springer, 1989, vol. 37. [Online]. Available: https://doi.org/10.1007/978-94-015-7860-2

[11] N. Srinivas, A. Krause, S. Kakade, and M. Seeger, “Gaussian process optimization in the bandit setting: no regret and experimental design,” in *Proceedings of the 27th International Conference on International Conference on Machine Learning*, ser. ICML’10. Madison, WI, USA: Omnipress, 2010, p. 1015–1022.

[12] C. E. Shannon, “A mathematical theory of communication,” *The Bell System Technical Journal*, vol. 27, no. 3, pp. 379–423, 1948.

[13] S. Kullback, *Information Theory and Statistics.* Mineola, New York: Dover Publications, 1968.

[14] C. P. Robert and G. Casella, *Monte Carlo Statistical Methods.* Springer, 2004.

[15] T. P. Minka, "Expectation propagation for approximate bayesian inference," in *Proceedings of the 17th Conference on Uncertainty in Artificial Intelligence (UAI).* San Francisco, CA, USA: Morgan Kaufmann, 2001, pp. 362–369.

[16] Z. Wang, S. Jegelka, and J. E. Gonzalez, "Max-value entropy search for efficient bayesian optimization," in *International Conference on Machine Learning*, 2017.

[17] P. Hennig and C. J. Schuler, "Entropy search for information-efficient global optimization," *Journal of Machine Learning Research*, vol. 13, no. Jun, pp. 1809–1837, 2012.

[18] J. M. Hernández-Lobato, M. W. Hoffman, and Z. Ghahramani, "Predictive entropy search for efficient global optimization of black-box functions," in *Advances in neural information processing systems*, 2014.