Multi-Scale Zero Order Optimization of Smooth Functions in an RKHS

Introduction
- Background
- Problem Setup

Algorithms
- `LP-GP-UCB` Algorithm
- `Heuristic` Algorithm

Main Results
- Theoretical Results
- Empirical Results

Black-box optimization

Consider the problem of finding a maximizer of a function $f : \mathcal{X} \to \mathbb{R}$.

Assumptions:

- f is not known explicitly; can only be accessed through a Zero Order Oracle.
- The observations of f are noisy.
- The function f is expensive to evaluate.

Goal: Design a sequential strategy of selecting query points to quickly reach a global optimizer of f.

Background

- This problem is intractable without any assumptions on f.
- Two common assumptions in literature are:
  1. f is a sample from a Gaussian Process $GP(0, K)$.
  2. f belongs to the RKHS of kernel K,
- Under both assumptions, GP can be used as a surrogate model for f.
- Prior work include algorithms such as GP-UCB.
- Large gaps between the existing upper and lower bounds on the performance.

Definitions

- **RKHS.** For a positive-definite kernel $K : \mathcal{X} \times \mathcal{X} \mapsto \mathbb{R}$, the RKHS of K, denoted by $\mathcal{H}_K$, is the *completion* of all finite linear combinations of the form $\sum_{j=1}^{m} c_j K(x_j, \cdot)$ for any $j \in \mathbb{N}$ and $\{x_1, \ldots, x_j\} \subset \mathcal{X}$.

- **Connection to GP.** Given some noisy observations $\{(x_i, y_i) \,:\, 1 \leq i \leq m\}$, computing the posterior mean of a GP using these samples is the same as constructing Kernel Ridge Regression (KRR) estimator of $f \in \mathcal{H}_K$.

- **Hölder Spaces** ($\mathcal{C}^{k,\alpha}$). Space of functions g whose k^{th} derivatives are Hölder continuous with exponent α.

  For $\mathcal{X} = \mathbb{R}$, this means $|g^{(k)}(x_1) - g^{(k)}(x_2)| \leq L|x_1 - x_2|^{\alpha}$ for all $x_1, x_2 \in \mathbb{R}$.

Problem Setup

- We assume that
  1. f has bounded norm in RKHS of K, i.e., $\|f\|_{\mathcal{H}_K} \leq B$
  2. f lies in the Hölder Space $\mathcal{C}^{k,\alpha}$.
- Observation model: $y = f(x) + \eta$ with $\eta \sim N(0, \sigma^2)$.
- $\mathcal{X}$ is a compact subset of $\mathbb{R}^D$
- Budget $= n$ evaluations
- Select queries $\{X_1, X_2, \ldots, X_n\}$ sequentially
- Performance measures:
  - Simple regret: $\mathcal{S}_n = f(x^*) - f(x_n^*)$
  - Cumulative regret: $\mathcal{R}_n = \sum_{t=1}^{n} f(x^*) - f(x_t)$

LP-GP-UCB Algorithm

Key Idea: Exploit the additional Hölder Smoothness information to augment the GP surrogate with Local Polynomial (LP) estimators to construct tighter UCBs.

Repeat the following steps for all times $t \geq 1$:

- Maintain a partition $\mathcal{P}_t = \{E_1, E_2, \ldots, E_{m_t}\}$.
- Use Local Polynomial (LP) Estimators along with global GP surrogate to construct UCB.
- Select query point $x_t \sim \mathtt{Unif}(E_t)$ where $E_t = \arg\max_{E \in \mathcal{P}_t} UCB(E)$.
- Update the partition $\mathcal{P}_t$ and the GP posterior.

Heuristic Algorithm

The `LP-GP-UCB` has some drawbacks in practical applications:

1. High memory requirements: $\Omega\left(2^D\right)$.
2. Works in the large n regime: $n = \Omega\left((k+2)^D\right)$.

We propose a `Heuristic` algorithm to address these issues.

Repeat for $t \geq 1$:

- Fit a Regression Tree to the data observed (local estimates).

  The leaves form a partition $\mathcal{P}_t = \{E_1, \ldots, E_{m_t}\}$ of $\mathcal{X}$.
- Fit a GP to the data observed (global model).
- Combine the two to construct a UCB.
- Select E_t with largest UCB value.
- Observe $y = f(x_t) + \eta$ at $x_t \sim \texttt{Unif}(E_t)$.

Theoretical Results

For Matérn kernels with $\nu > 0$, we can show an embedding of the RKHS into the space $\mathcal{C}^{k,\alpha}$ for $k = \lceil \nu - 1 \rceil$ and $\alpha = \nu - k$.

- For Matérn kernels, the `LP-GP-UCB` algorithms achieves uniformly tighter bounds on both $\mathcal{S}_n$ and $\mathcal{R}_n$, for all $\nu > 0$.
- In particular, the bounds on $\mathcal{S}_n$ (resp. $\mathcal{R}_n$) match the algorithm independent lower bounds for $\nu \leq D(D+1)$ (resp. $\nu \leq 1$).
- Besides Matérn kernels, we also obtain the first explicit in n regret bounds for some other important kernels such as Rational-Quadratic and Gamma-Exponential.

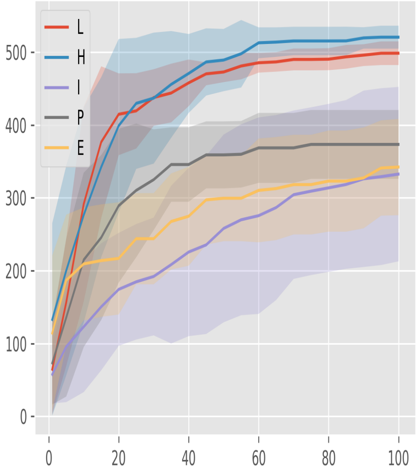
Empirical Results-I

We use the 2-dim Branin function (g_B) to construct an objective function (f) on $\mathbb{R}^8$ as follows:

$$f(x) = \sum_{i=1}^{4} c_i g_B \left(x[2i-1:2i]\right),$$

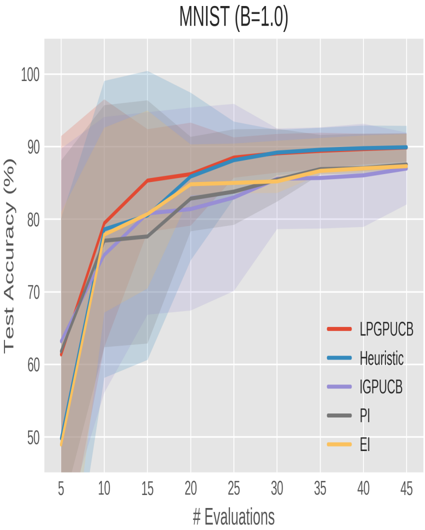
where $c_1 = 1.0$ and $c_i = 0.1$ for $i = 2, 3, 4$.

Informally, f has 2 *active* dimensions and an *ambient* dimension of 8.



Empirical Results-II

- Hyperparameter tuning for a CNN with 2 conv layers and 2 fully-connected layers.
- The tunable parameters are `batch_size`, `learning_rate`, `kernel_size`$\times 2$ and `hidden_layer_size`.
- The objective function value is the training error.



Empirical Results-III

Task / Algo.	Branin24	Goldstein24	Hartman6	MNIST
LP-GP-UCB			2.56 ± 0.41	$90.18\% \pm 1.43$
Heuristic	-25.28 ± 2.75	-5.84 ± 2.16	2.47 ± 0.35	$90.82\% \pm 1.71$
πGPUCB			2.29 ± 0.37	$88.74\% \pm 1.98$
IGPUCB	-284.16 ± 70.43	-93.19 ± 68.49	1.90 ± 0.77	$85.03\% \pm 10.98$
GP-EI	-252.80 ± 33.81	-58.07 ± 21.85	3.26 ± 0.23	$88.89\% \pm 1.67$
GP-PI	-265.88 ± 34.77	-66.02 ± 25.31	3.24 ± 0.26	$86.69\% \pm 2.96$

Table: Highest function value found by the optimization algorithms.