

Gaussian Process Optimization in the Bandit Setting: No Regret and Experimental Design

Niranjan Srinivas, Andreas Krause[†], Sham Kakade^{††}, Matthias Seeger^{†††}

[†]: California Institute of Technology

^{††}: University of Pennsylvania

^{†††}: Saarland University

December 27, 2022

目次

- 1 はじめに
- 2 ベイズ最適化
- 3 GP-UCB
- 4 実験
- 5 まとめ

1 はじめに

導入

ベイズ最適化

ガウス過程

バンディット問題

2 ベイズ最適化

3 GP-UCB

4 実験

5 まとめ

- 計算機科学や統計学において、最適化問題は非常に重要であることを強調する
- ガウス過程を用いた最適化は、複雑で非線形な関数を扱う場合に有用であることを紹介する
- バンディット設定において、最適な決定をすることが重要であることを紹介する
- 今までにバンディット設定でのガウス過程最適化については、リグレット制限が証明されていなかった
- 本論文では、バンディット設定におけるガウス過程最適化において、リグレット制限を証明することで、最適な決定をするアルゴリズムを提案する

- 序論として，ベイズ最適化について説明する (前提知識の提供)

- 序論として，ガウス過程について説明する (前提知識の提供)

- 序論として、バンディット問題について説明する (前提知識の提供)
- バンディット設定とは、決定をする際に報酬を得ることができるが、そのためには、その決定をする前にその詳細を知ることができない設定を指す
- このような設定では、報酬を最大化することを目的として、決定をする場所を最適に選択することが重要になる
- バンディット問題の具体例: 投資をする場合は、商品を購入する場合など. このような場面では、最適な決定をすることで、最大の報酬を得ることができるため、重要である.

1 はじめに

2 **ベイズ最適化**

ベイズ最適化 1

ベイズ最適化 2

3 GP-UCB

4 実験

5 まとめ

ブロック

a

- テスト

- テスト

1 はじめに

2 ベイズ最適化

3 GP-UCB

GP-UCB1

GP-UCB2

4 実験

5 まとめ

- GP-UCB は、バンディット設定でのガウス過程最適化において用いられるアルゴリズムである。
- バンディット設定では、報酬を最大化することを目的として、決定をする場所を最適に選択することが重要である。
- GP-UCB はこのような場合に、ガウス過程を用いて最適な決定をするためのアルゴリズムである
-

- テスト

1 はじめに

2 ベイズ最適化

3 GP-UCB

4 実験

実験 1

実験 2

5 まとめ

- テスト

- テスト

1 はじめに

2 ベイズ最適化

3 GP-UCB

4 実験

5 **まとめ**

まとめ 1

まとめ 2

- We analyze GP-UCB, an intuitive algorithm for GP optimization, when the function is either sampled from a known GP, or has low RKHS norm.
- We bound the cumulative regret for GP-UCB in terms of the information gain due to sampling, establishing a novel connection between experimental design and GP optimization.
- By bounding the information gain for popular classes of kernels, we establish sublinear regret bounds for GP optimization for the first time. Our bounds depend on kernel choice and parameters in a fine-grained fashion.
- We evaluate GP-UCB on sensor network data, demonstrating that it compares favorably to existing algorithms for GP optimization.

- テスト