HEBO: Heteroscedastic Evolutionary BayesianOptimisation

论文主要信息

- 标题：HEBO: Heteroscedastic Evolutionary BayesianOptimisation
- 会议：NIPS
- 作者：Alexander I. Cowen-Rivers、Wenlong Lyu
- 机构：华为诺亚实验室
- 论文：https://arxiv.org/pdf/2012.03826v1.pdf
- 代码：https://github.com/huawei-noah/noah-research/tree/master/BO/HEBO

文章概要

摘要

本文HEBO算法赢得NeurIPS 2020 BBO优化比赛冠军，对surrogate model、acquisition function的修改（non-conventional）超越Bayesmark包的baseline，通过对比消融实验说明HEBO的成功

研究内容

介绍

贝叶斯优化

- 序列决策方法找函数 $f: \mathcal{X} \rightarrow \mathbb{R}$ 的全局最优解
- 使用GP建模 f ，利用观测数据 $\mathcal{D}_i$ 更新posterior： $p(f(\cdot)|\mathcal{D}_i)$

$$\min_{\boldsymbol{\theta}, \sigma_{\text{noise}}} \mathcal{J}\left(\boldsymbol{\theta}, \sigma_{\text{noise}}\right) = \frac{1}{2} \log \det\left(\boldsymbol{C}_{\boldsymbol{\theta}}^{(i)}\right) + \frac{1}{2}\left(\boldsymbol{y}_{1:n_i} - m\left(\boldsymbol{x}_{1:n_i}\right)\right)^{\top} \boldsymbol{C}_{\boldsymbol{\theta}}^{(i),-1}\left(\boldsymbol{y}_{1:n_i} - m\left(\boldsymbol{x}_{1:n_i}\right)\right) + \frac{n_i}{2} \log 2\pi$$

$$\text{with } C_{\boldsymbol{\theta}}^{(i)} = \boldsymbol{K}_{\boldsymbol{\theta}}^{(i)} + \sigma_{\text{noise}}^2 \boldsymbol{I}$$

- Acquisition function采样下一批数据
  - Expected Improvement（EI）

$$\alpha_{\mathrm{q-EI}}\left(\boldsymbol{x}_{1:q} \mid \mathcal{D}_i\right)=\mathbb{E}_{\boldsymbol{f}\left(\boldsymbol{x}_{1:q}\right)\mid\mathcal{D}_i,\boldsymbol{\theta}}\left[\max_{j\in 1:q}\left\{\operatorname{ReLU}\left(\boldsymbol{f}\left(\boldsymbol{x}_{1:q}\right)-f\left(\boldsymbol{x}_i^{+}\right)\mathbf{1}_q\right)\right\}\right]$$

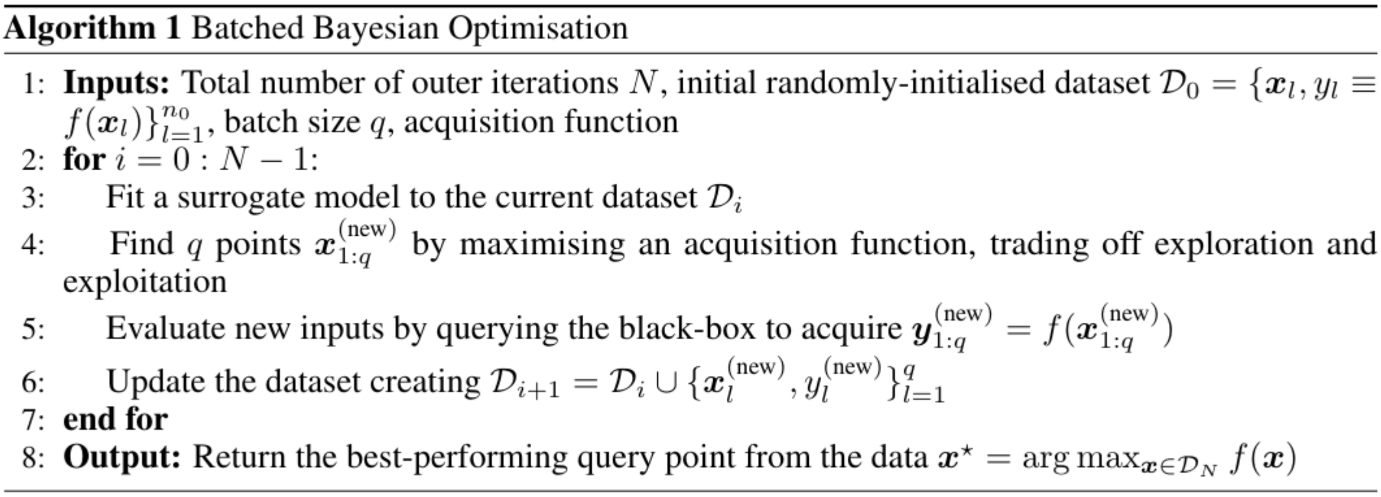
- Probability of Improvement

$$\alpha_{\mathrm{q-PI}}\left(\boldsymbol{x}_{1:q} \mid \mathcal{D}_i\right)=\mathbb{E}_{\boldsymbol{f}\left(\boldsymbol{x}_{1:q}\right)\mid\mathcal{D}_i,\boldsymbol{\theta}}\left[\max_{j\in 1:q}\left\{1\left\{\boldsymbol{f}\left(\boldsymbol{x}_{1:q}\right)-f\left(\boldsymbol{x}_i^{+}\right)\mathbf{1}_q\right\}\right\}\right]$$

- Upper Confidence Bound（UCB）

$$\alpha_{\mathrm{q-UCB}}\left(\boldsymbol{x}_{1:q} \mid \mathcal{D}_i\right)=\mathbb{E}_{\boldsymbol{f}\left(\boldsymbol{x}_{1:q}\right)\mid\mathcal{D}_i,\boldsymbol{\theta}}\left[\max_{j\in 1:q}\left\{\boldsymbol{\mu}_i\left(\boldsymbol{x}_{1:q};\boldsymbol{\theta}\right)+\sqrt{\beta\pi/2}\left|\boldsymbol{\gamma}_i\left(\boldsymbol{x}_{1:q};\boldsymbol{\theta}\right)\right|\right\}\right]$$
$$\text{where } \gamma_i\left(\boldsymbol{x}_{1:q};\boldsymbol{\theta}\right)=\boldsymbol{f}\left(\boldsymbol{x}_{1:q}\right)-\boldsymbol{\mu}_i\left(\boldsymbol{x}_{1:q};\boldsymbol{\theta}\right)$$

整体算法流程



Algorithm 1 Batched Bayesian Optimisation

```
Inputs: Total number of outer iterations $N$, initial randomly-initialised dataset $\mathcal{D}_0=\{\boldsymbol{x}_l, y_l \equiv f(\boldsymbol{x}_l)\}_{l=1}^{n_0}$, batch size $q$, acquisition function
for $i=0:N-1$:
    Fit a surrogate model to the current dataset $\mathcal{D}_i$
    Find $q$ points $\boldsymbol{x}_{1:q}^{(\mathrm{new})}$ by maximising an acquisition function, trading off exploration and exploitation
    Evaluate new inputs by querying the black-box to acquire $\boldsymbol{y}_{1:q}^{(\mathrm{new})}=f(\boldsymbol{x}_{1:q}^{(\mathrm{new})})$
    Update the dataset creating $\mathcal{D}_{i+1}=\mathcal{D}_i \cup \{\boldsymbol{x}_l^{(\mathrm{new})}, y_l^{(\mathrm{new})}\}_{l=1}^{q}$
end for
Output: Return the best-performing query point from the data $\boldsymbol{x}^{\star}=\arg\max_{\boldsymbol{x}\in\mathcal{D}_N} f(\boldsymbol{x})$
```

Surrogate Model 改进方法

困难：heteroscedasticity、 non-stationarity data

Output： 采用 power transformation（sklearn 包）使得 y 变换后等方差且相互独立

- box-cox（要求非负）
- yeo-johnson

- **Input**：input warped Gaussian process（GPy包）使得核函数稳定
  - 理解：对输入空间变换，使核函数更加稳定，比如RBF
- **Gaussian Process & Kernels**：选择使用俩kernel之和作为GP的核函数族（sample efficiency? ）
  - $k_{\boldsymbol{\theta}_1,\boldsymbol{\theta}_2}^{(\mathrm{HEBO})}\left(\boldsymbol{x},\boldsymbol{x}^{\prime}\right)=k_{\boldsymbol{\theta}_1}^{(\mathrm{Linear})}\left(\boldsymbol{x},\boldsymbol{x}^{\prime}\right)+k_{\boldsymbol{\theta}_2}^{(\mathrm{Matern32})}\left(\boldsymbol{x},\boldsymbol{x}^{\prime}\right)$

- **Stochastic mean function**：GP的后验均值加入noise-dependent（超参控制？）来增加exploration
  - $m^{(\text{HEBO})}(\cdot)=m(\cdot)+\boldsymbol{\xi}\sigma_{\text{noise}}^{2}$

Acquisitions改进方法

不同的acquisitions在某些时候会conflict => 多目标

Pareto front：if no objective can be improved without sacrificing at least one other objective

（多目标优化的结果比单目标优化的结果好，单目标优化牺牲了其他的目标）

定义Acuisition function：

$$\min_{\boldsymbol{x}\in\mathcal{X}}\left(-\alpha_{\mathrm{q-EI}}\left(\boldsymbol{x}\mid\mathcal{D}_{i}\right),-\alpha_{\mathrm{q-PI}}\left(\boldsymbol{x}\mid\mathcal{D}_{i}\right),\alpha_{\mathrm{q-UCB}}\left(\boldsymbol{x}\mid\mathcal{D}_{i}\right)\right)$$

multi-objective acquisition ensemble algorithm (MACE)

NSGA-II 优化器， pymoo包

$$\alpha_{\mathrm{q-EI}}\left(\boldsymbol{x}\mid\mathcal{D}_{i}\right)=\sigma_{i}(\boldsymbol{x};\boldsymbol{\theta})\left(\boldsymbol{z}_{i}\Phi\left(\boldsymbol{z}_{i}\right)+\phi\left(\boldsymbol{z}_{i}\right)\right)\text{ with }\boldsymbol{z}_{i}=\frac{\tau-\boldsymbol{\mu}_{i}(\boldsymbol{x};\boldsymbol{\theta})}{\sigma_{i}(\boldsymbol{x};\boldsymbol{\theta})}$$

problem：优化的时候初始点可能接近0,梯度也接近0

a. 优化对数目标（导致数值不稳定）

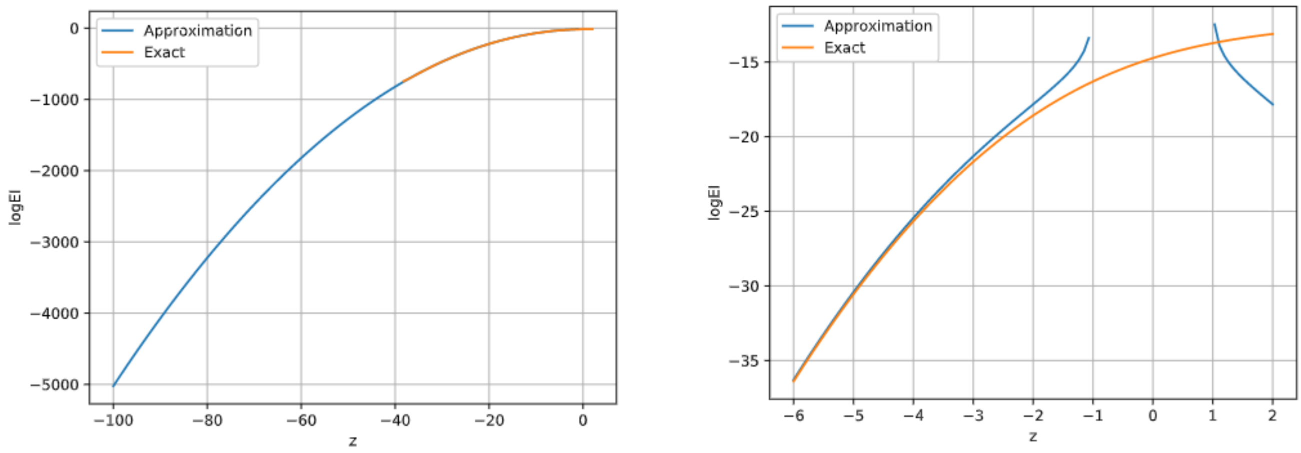
$$\log\alpha_{\mathrm{q-EI}}\left(\boldsymbol{x}\mid\mathcal{D}_{i}\right)=\log\sigma_{i}(\boldsymbol{x};\boldsymbol{\theta})+\log\phi\left(\boldsymbol{z}_{i}\right)+\log\left(1+\frac{\boldsymbol{z}_{i}\Phi\left(\boldsymbol{z}_{i}\right)}{\phi\left(\boldsymbol{z}_{i}\right)}\right)$$

b. Log Approximation

使用**Log Approximation**：

$$\lim_{\boldsymbol{z}_{i}\to-\infty}\log\alpha_{\mathrm{q-EI}}\left(\boldsymbol{x}\mid\mathcal{D}_{i}\right)=\log\sigma_{i}(\boldsymbol{x};\boldsymbol{\theta})-\frac{1}{2}\boldsymbol{z}_{i}^{2}-\log\left(\boldsymbol{z}_{i}^{2}-1\right)-\frac{1}{2}\log(2\pi),\text{ where }\boldsymbol{z}_{i}=\frac{\tau-\boldsymbol{\mu}_{i}(\boldsymbol{x};\boldsymbol{\theta})}{\sigma_{i}(\boldsymbol{x};\boldsymbol{\theta})}$$

当 $\boldsymbol{z}_{i}<-6$ 使用近似



(a)

(b)

Figure 2: (a-b) Comparison between the exact $\log \alpha_{\text{q-EI}}(\boldsymbol{x}_{1:q}|\mathcal{D}_i)$ and the proposed logarithm approximation, note that when $z < -40$, exact evaluation returns NaN while the approximation can still be calculated.

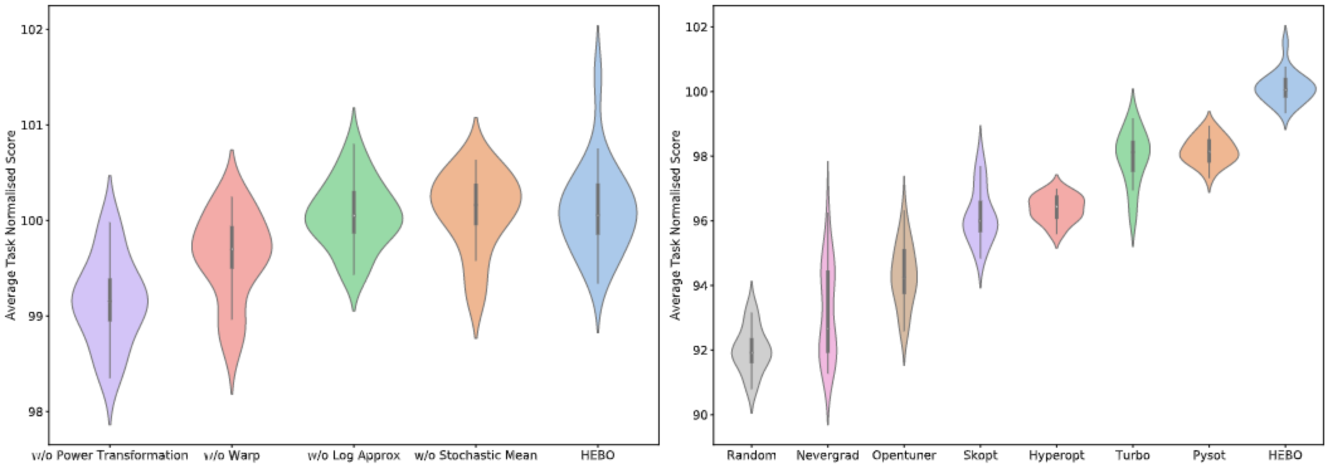
实验

数据集

- Bayesmark **Offline** Datasets
    - https://github.com/uber/bayesmark#selecting-the-data-set
    - https://scikit-learn.org/stable/datasets/toy_dataset.html
- Bayesmark baseline：https://github.com/rdturnermtl/bbo_challenge_starter_kit/

实验

> 108 tasks（6 data sets × 9 classifiers × 2 metrics(分类/回归)）
>
> 每个task用20个random seeds重复
>
> $q = 1$



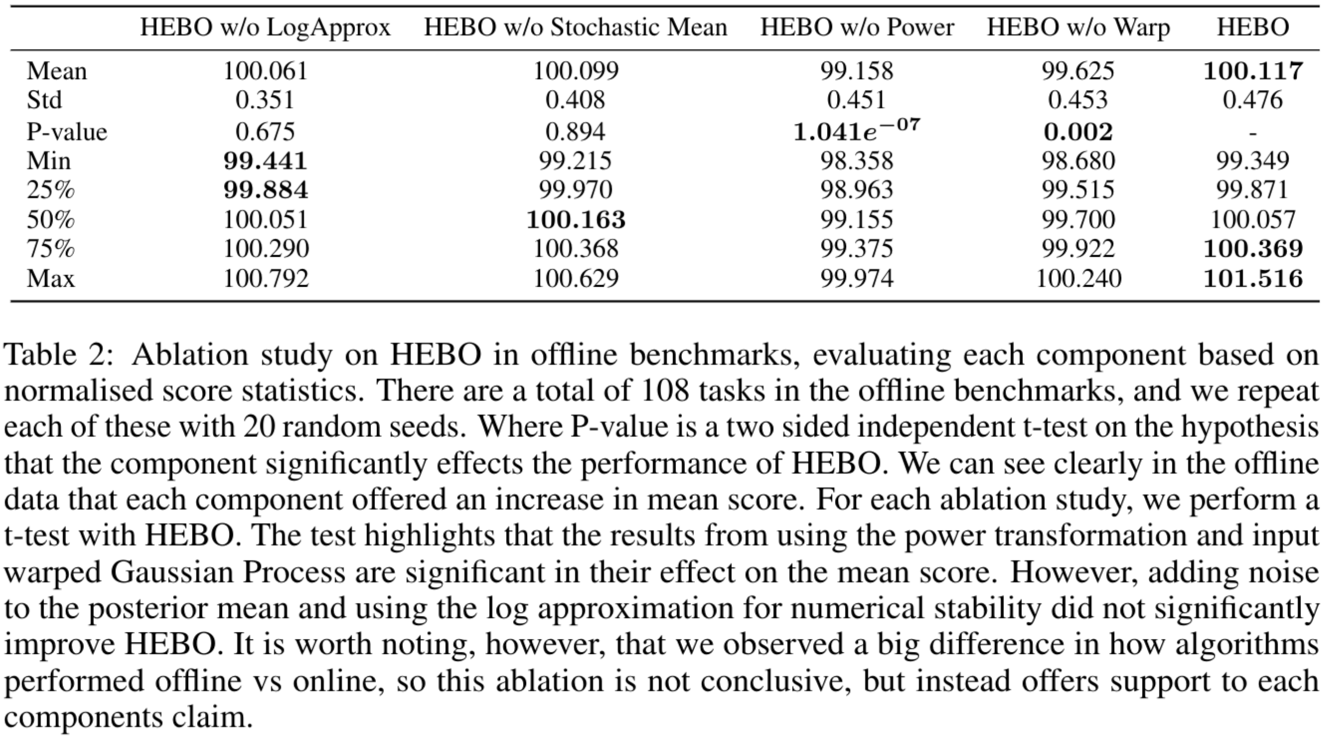
(a) (b)

Figure 1: All algorithms in the plots are evaluated on the full `Bayesmark` Offline Dataset of 108 tasks and each task repeated with 20 random seeds. (a) Ablation Study of HEBO on Bayesmark Offline Dataset. We see that when all components are combined, we see a more appealing upper quartile distribution which reaches a much higher maximum scores than other ablations were able to, hinting towards a positively synergistic combination of algorithm components was found. (b) Summary of HEBO vs all available baselines. We see substantial improvements when using HEBO against all baselines, in many cases with HEBO also producing tighter distribution of scores on the lower quartile, and a high distribution on the upper quartile. This distribution of quartiles suggests that worst case we know it should roughly perform near the mean, and best case we could get a result significantly higher than the mean. This, paired with the fact that our mean performance is above satisfactory, give us much confidence in applying this method to a wide-range of BO tasks. Notice, as we significantly outperformed all baselines, similarly we see that Pysot and Turbo significantly outperform all *other* baselines.

	Hyperopt	Opentuner	Random	Nevergrad	Pysot	Skopt	Turbo	HEBO
Mean	96.375	94.317	92.004	93.196	98.177	96.175	97.951	**100.117**
Std	0.407	0.981	0.699	1.457	0.417	0.853	0.847	0.4765
P-value	$\mathbf{7.9e^{-26}}$	$\mathbf{6.9e^{-20}}$	$\mathbf{7.3e^{-31}}$	$\mathbf{3.8e^{-16}}$	$\mathbf{3.8e^{-16}}$	$\mathbf{1.3e^{-17}}$	$\mathbf{5.0e^{-11}}$	-

Table 1: `Bayesmark` Offline Dataset Summary of HEBO vs all available benchmarks on 108 tasks, each task repeated with 20 random seeds. Where P-value is a two-sided test for the null hypothesis that an algorithm has the same expected value as HEBO: p-values are highlight in bold if this null hypothesis is rejected with 95% significance. We found all methods to be significantly worse than HEBO.

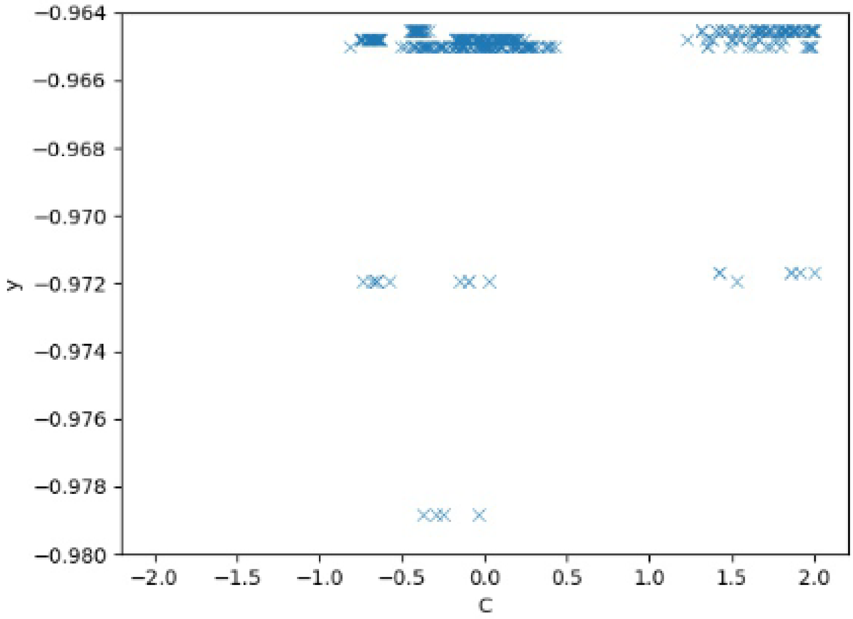
- （a）消融实验
    - 组合所有算法组件有显著高的最高score => 算法组件间发挥积极协同作用
- （b）与Bayesmark所有baseline对比
    - HEBO下四分位数分布集中 => 在最坏的情况下接近均值
    - HEBO上四分位数分布细长 => 在最好的情况下明显高于均值
    - Pysot和Turbo效果比其他baseline好



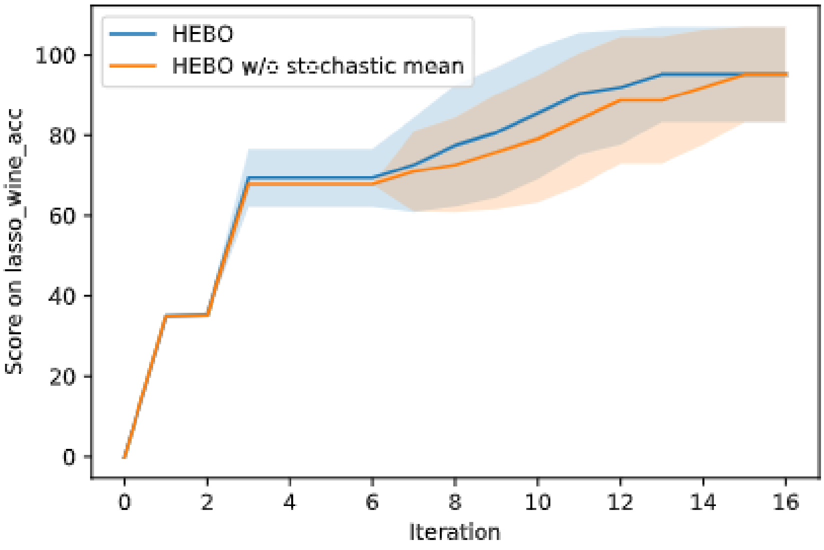
	HEBO w/o LogApprox	HEBO w/o Stochastic Mean	HEBO w/o Power	HEBO w/o Warp	HEBO
Mean	100.061	100.099	99.158	99.625	**100.117**
Std	0.351	0.408	0.451	0.453	0.476
P-value	0.675	0.894	$\mathbf{1.041e^{-07}}$	**0.002**	-
Min	**99.441**	99.215	98.358	98.680	99.349
25%	**99.884**	99.970	98.963	99.515	99.871
50%	100.051	**100.163**	99.155	99.700	100.057
75%	100.290	100.368	99.375	99.922	**100.369**
Max	100.792	100.629	99.974	100.240	**101.516**

Table 2: Ablation study on HEBO in offline benchmarks, evaluating each component based on normalised score statistics. There are a total of 108 tasks in the offline benchmarks, and we repeat each of these with 20 random seeds. Where P-value is a two sided independent t-test on the hypothesis that the component significantly effects the performance of HEBO. We can see clearly in the offline data that each component offered an increase in mean score. For each ablation study, we perform a t-test with HEBO. The test highlights that the results from using the power transformation and input warped Gaussian Process are significant in their effect on the mean score. However, adding noise to the posterior mean and using the log approximation for numerical stability did not significantly improve HEBO. It is worth noting, however, that we observed a big difference in how algorithms performed offline vs online, so this ablation is not conclusive, but instead offers support to each components claim.

- power transformation 、 input warped Gaussian Proces对score有显著提升效果
- Offline vs Online，算法表现上有较大差异，这里仅用offline数据来说明改进组件的效果

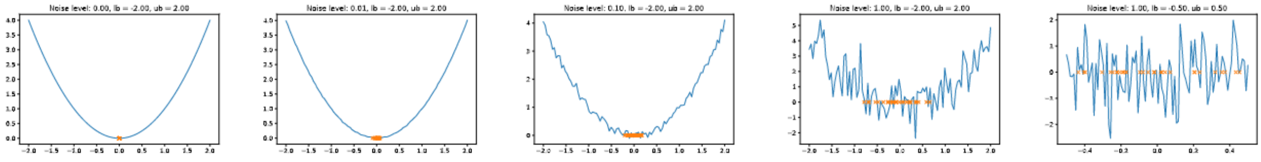


(a)



(b)

在某些时候random search更有效



(a) (b) (c) (d) (e)

Figure 5: $f(x) = x^2$ Synthetic black-box function experiment to demonstrate the optimised points (orange crosses) that a genetic algorithm arrives at when varying noise levels are introduced, which simulate the varying likelihood noise values that effect the posterior mean prediction. We can clearly see the correlation of, as noise increases as does the fine spread of optimised. It is this principle that allows HEBO to achieve a coarse-to-fine search

在后验均值上加入噪声的幅度可以理解为在exploration和exploitation的trade-off