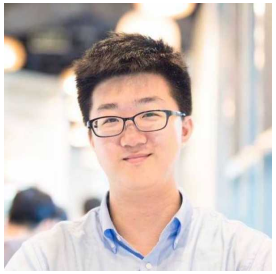
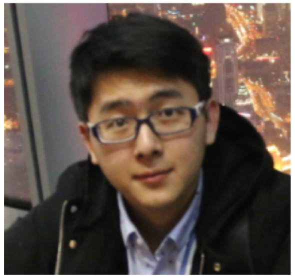
A General Recipe for Likelihood-free Bayesian Optimization

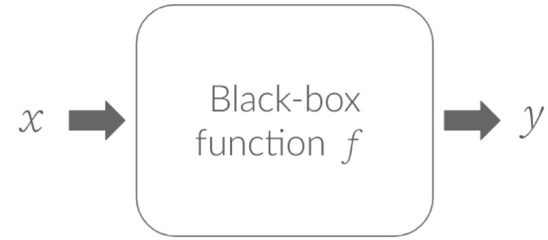



Jiaming Song* Lantao Yu* Willie Neiswanger Stefano Ermon

Project website: https://lfbo-ml.github.io/

Stanford University

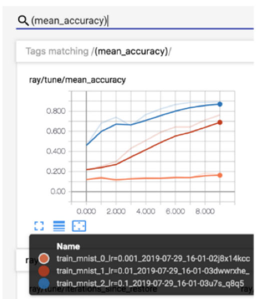
Black-box Global Optimization



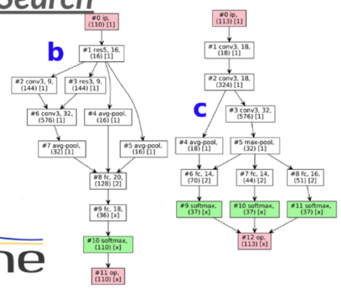
Suppose we have a noisy *"black-box"* function f.

Goal: estimate the location of global optima of f.

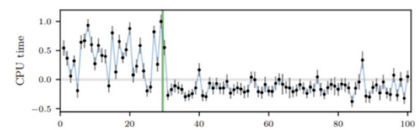
Hyperparameter Opt & Neural Architecture Search

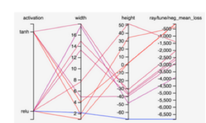


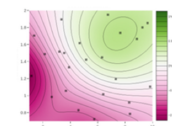
tune

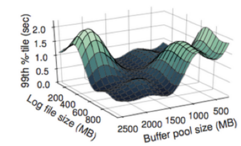


Systems Auto-tuning

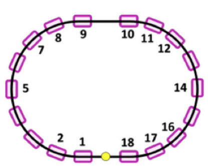




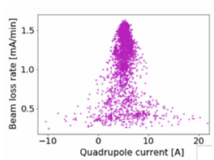




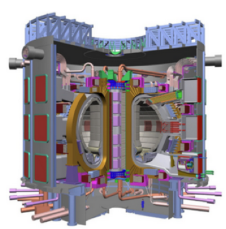
Optimizing Laboratory Equipment & Machines



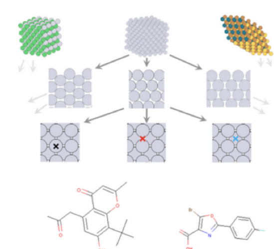
(a) Layout of SPEAR3



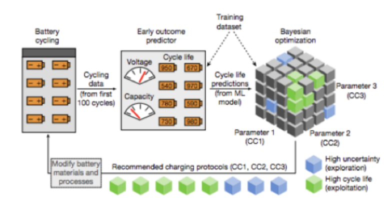
(b) Beam loss rate



Materials Discovery & Protocols

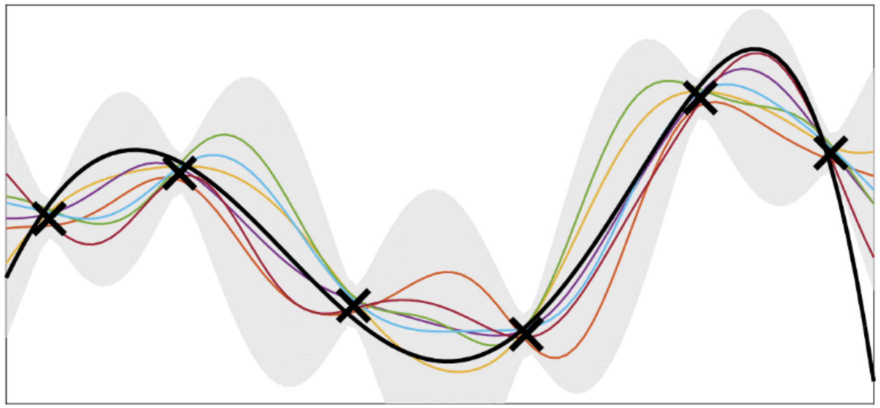


(a) QED 0.92083 (b) QED 0.92145



Bayesian Optimization

Probabilistic model of f



From past observations:

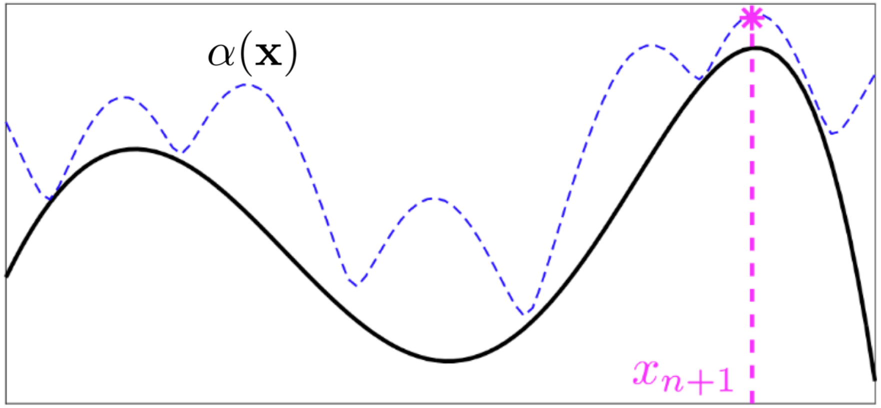
$$\mathcal{D}_n = \{(x_i, y_i)\}_{i=1}^n$$

Define the model as:

$$p(y|\mathbf{x}, \mathcal{D}_n)$$

Most popular one is Gaussian Process

Use model to choose queries



Construct acquisition function from model, then find its maximum to query.

$$\mathbf{x}_{n+1} = \arg\max \alpha(\mathbf{x})$$

Acquisition Functions

Many acquisition functions are defined as expected "utility" over the model:

$$\alpha(\mathbf{x}) = \mathbb{E}_{p(y|\mathbf{x};\mathcal{D}_n)}[u(y)]$$

Probability of improvement (PI):

$$u(y) = \mathbb{I}(y > \tau)$$

Indicator of y being over some threshold τ

Expected improvement (EI)

$$u(y) = \max(y - \tau, 0)$$

How much is y over some threshold τ

Expectations often have analytical form for Gaussian processes (GPs)

Drawbacks of BO with Gaussian Processes

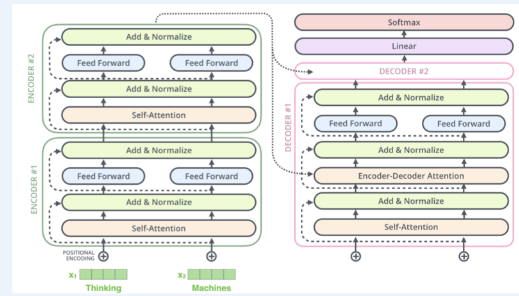
$O(n^3)$

for basic GP inference.

$O(n)$

for sparse GP inference.

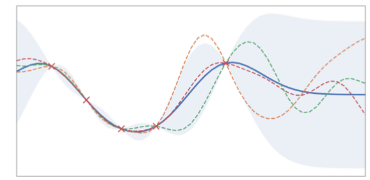
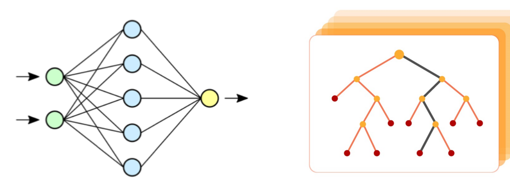
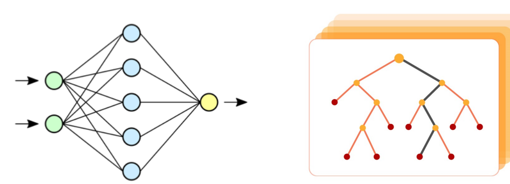
Scalability



Expressiveness

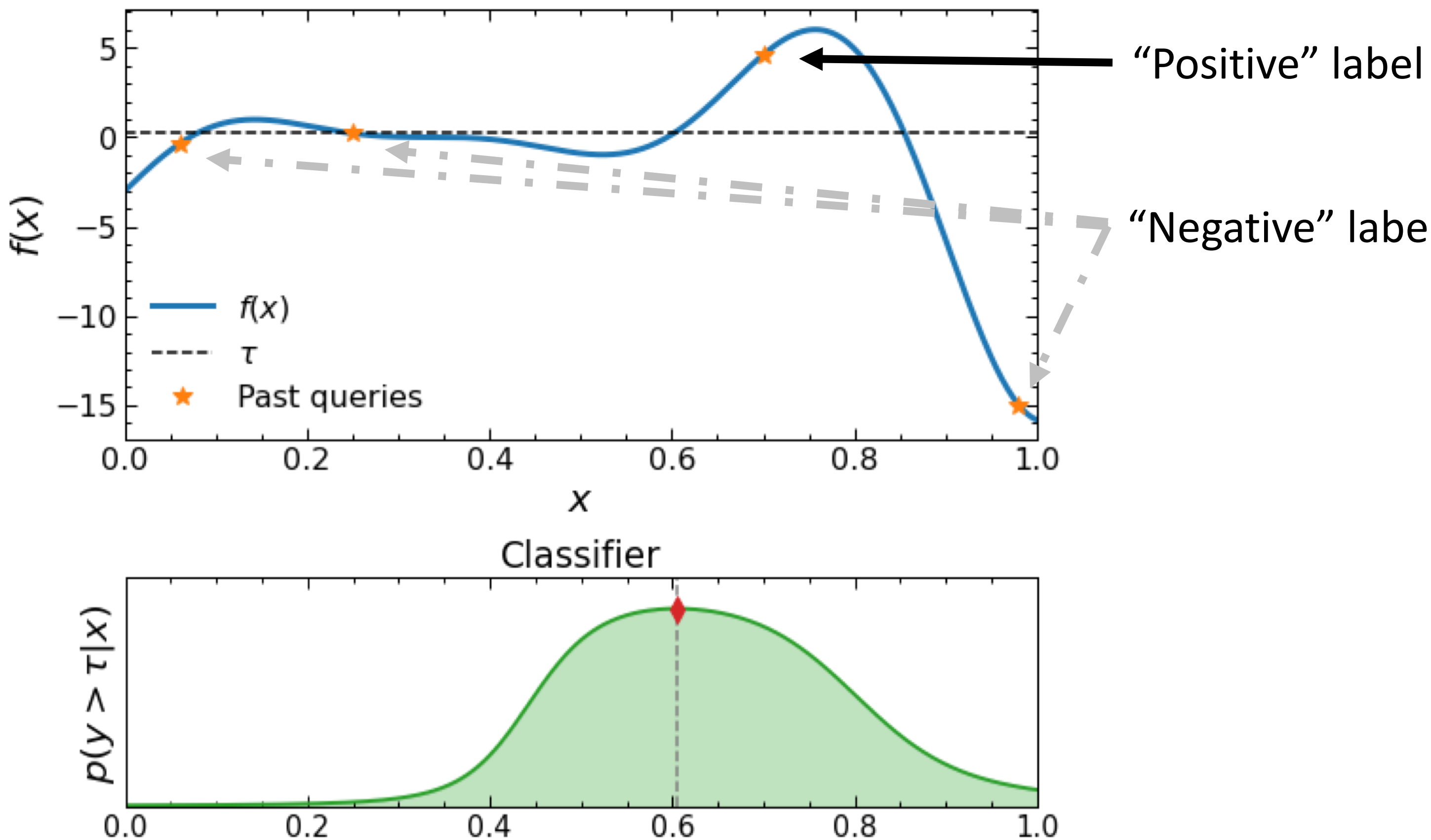
- In BO, query only depends on acquisition function!
- Can we do this without a separate probabilistic model?

Overview

	Utility function	*Model*
Analytical expected utility	PI, EI, and some others	 model with tractable probabilities (e.g., GPs)
A Classifier-based Approach	PI	 deterministic model
LFBO (ours)	All non-negative functions	 deterministic model

Bayesian Optimization via classification

Prior work has proposed using classifiers for Bayesian optimization:



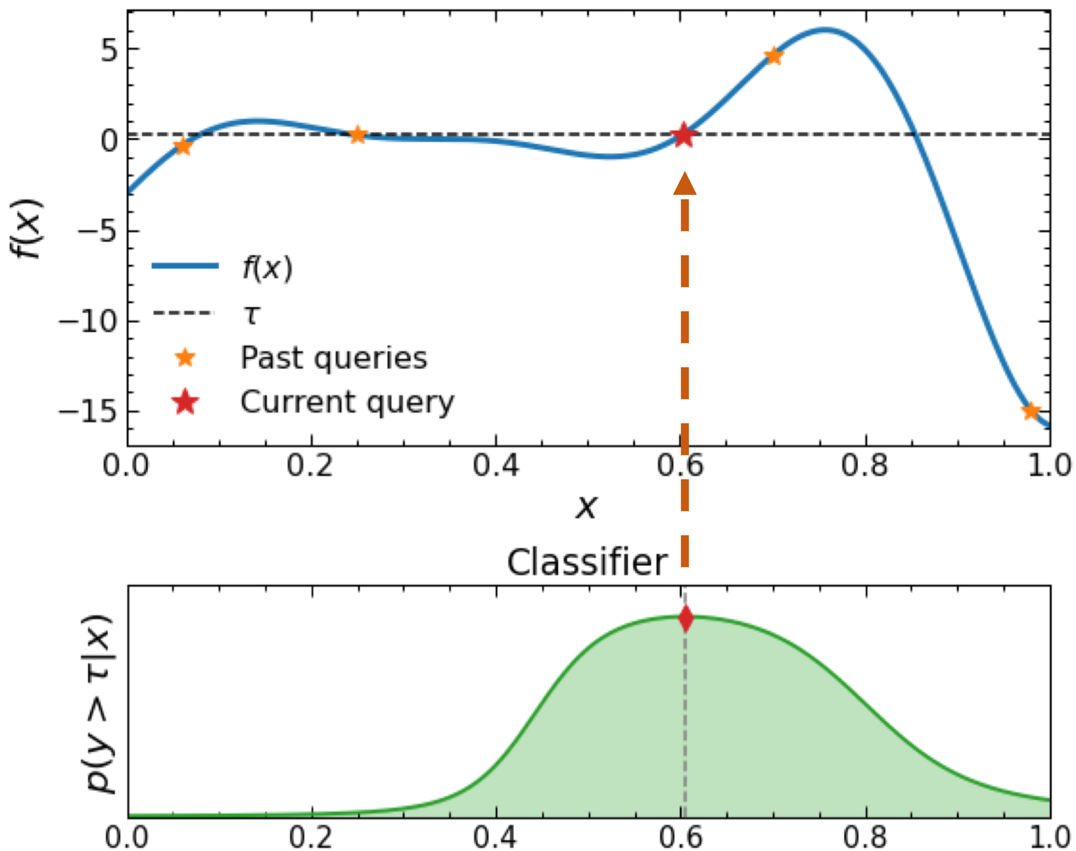
[1] **Algorithms for hyper-parameter optimization**. Bergstra, J., Bardenet, R., Bengio, Y. and Kégl, B. Advances Neural information processing systems (NeurIPS), 24. 2011

[2] **BORE: Bayesian optimization by density-ratio estimation**. Tiao, L.C., Klein, A., Seeger, M.W., Bonilla, E.V., Archambeau, C. and Ramos, F. International Conference on Machine Learning (ICML), PMLR. 2021.

Bayesian Optimization via classification

Prior work has proposed using classifiers for Bayesian optimization:

Step 1



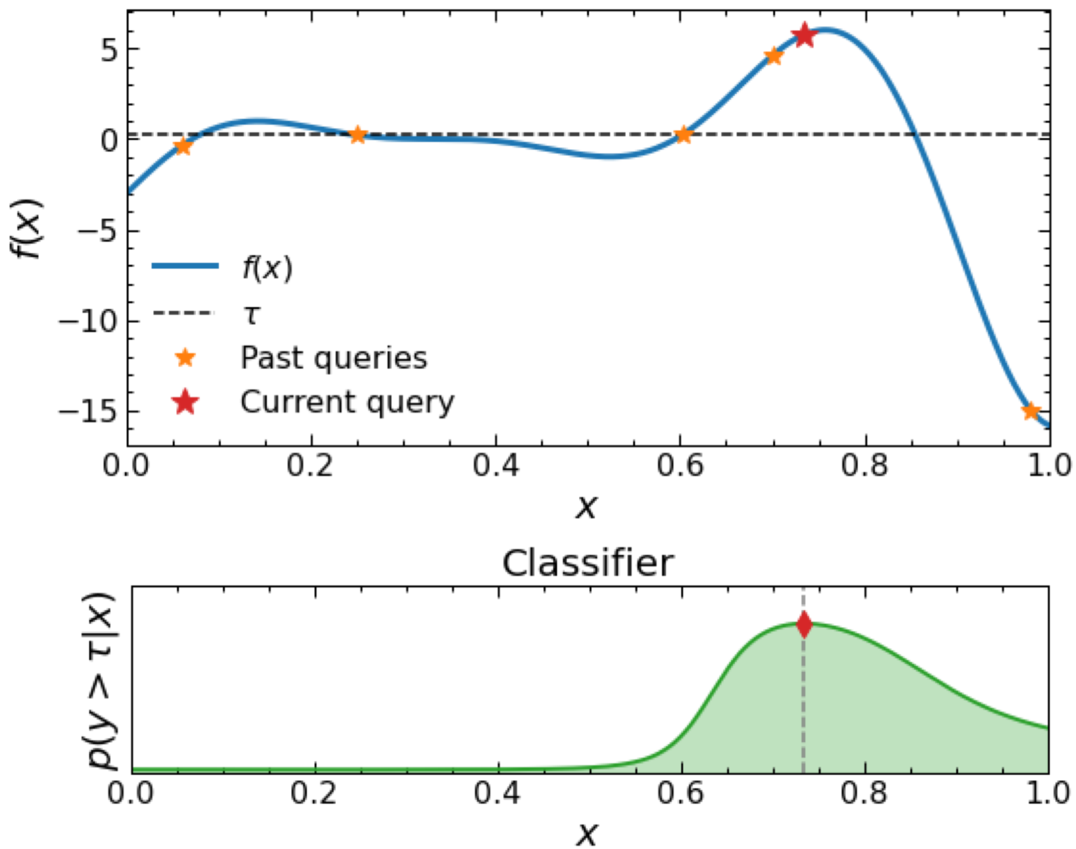
[1] **Algorithms for hyper-parameter optimization**. Bergstra, J., Bardenet, R., Bengio, Y. and Kégl, B. Advances Neural information processing systems (NeurIPS), 24. 2011

[2] **BORE: Bayesian optimization by density-ratio estimation**. Tiao, L.C., Klein, A., Seeger, M.W., Bonilla, E.V., Archambeau, C. and Ramos, F. International Conference on Machine Learning (ICML), PMLR. 2021.

Bayesian Optimization via classification

Prior work has proposed using classifiers for Bayesian optimization:

Step 2



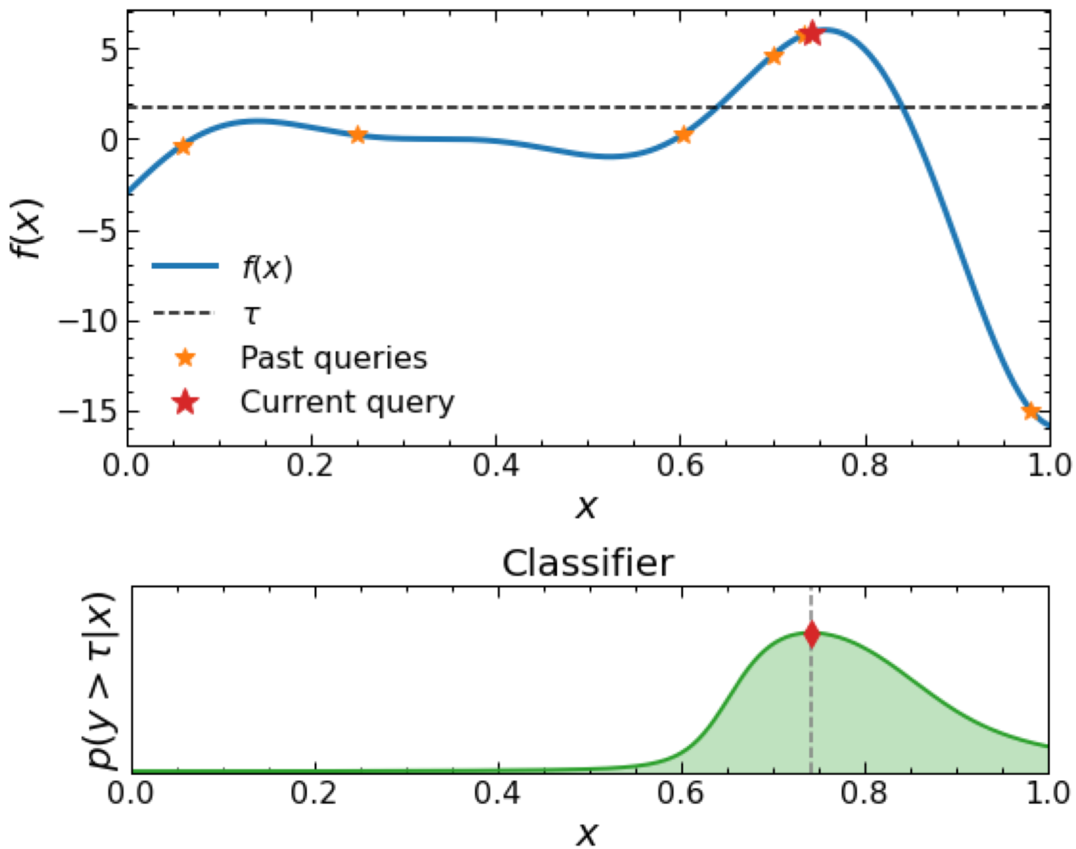
[1] **Algorithms for hyper-parameter optimization**. Bergstra, J., Bardenet, R., Bengio, Y. and Kégl, B. Advances Neural information processing systems (NeurIPS), 24. 2011

[2] **BORE: Bayesian optimization by density-ratio estimation**. Tiao, L.C., Klein, A., Seeger, M.W., Bonilla, E.V., Archambeau, C. and Ramos, F. International Conference on Machine Learning (ICML), PMLR. 2021.

Bayesian Optimization via classification

Prior work has proposed using classifiers for Bayesian optimization:

Step 3



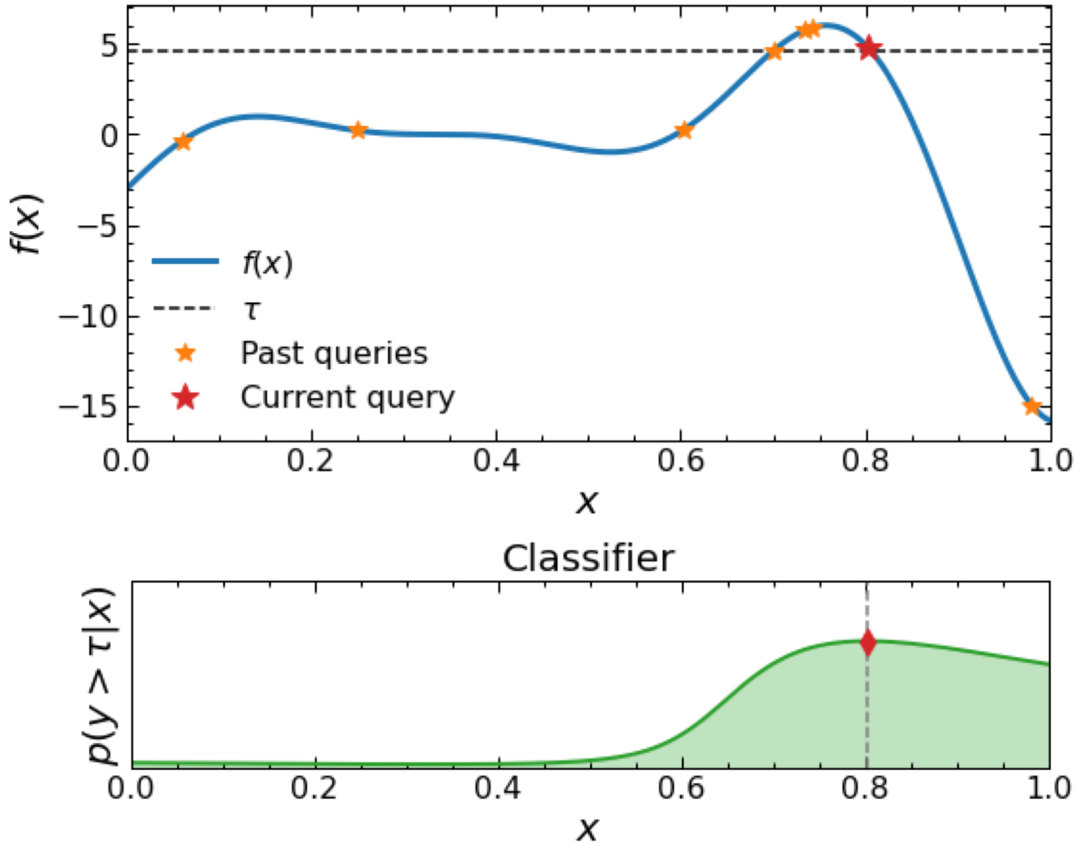
[1] **Algorithms for hyper-parameter optimization**. Bergstra, J., Bardenet, R., Bengio, Y. and Kégl, B. Advances Neural information processing systems (NeurIPS), 24. 2011

[2] **BORE: Bayesian optimization by density-ratio estimation**. Tiao, L.C., Klein, A., Seeger, M.W., Bonilla, E.V., Archambeau, C. and Ramos, F. International Conference on Machine Learning (ICML), PMLR. 2021.

Bayesian Optimization via classification

Prior work has proposed using classifiers for Bayesian optimization:

Step 4



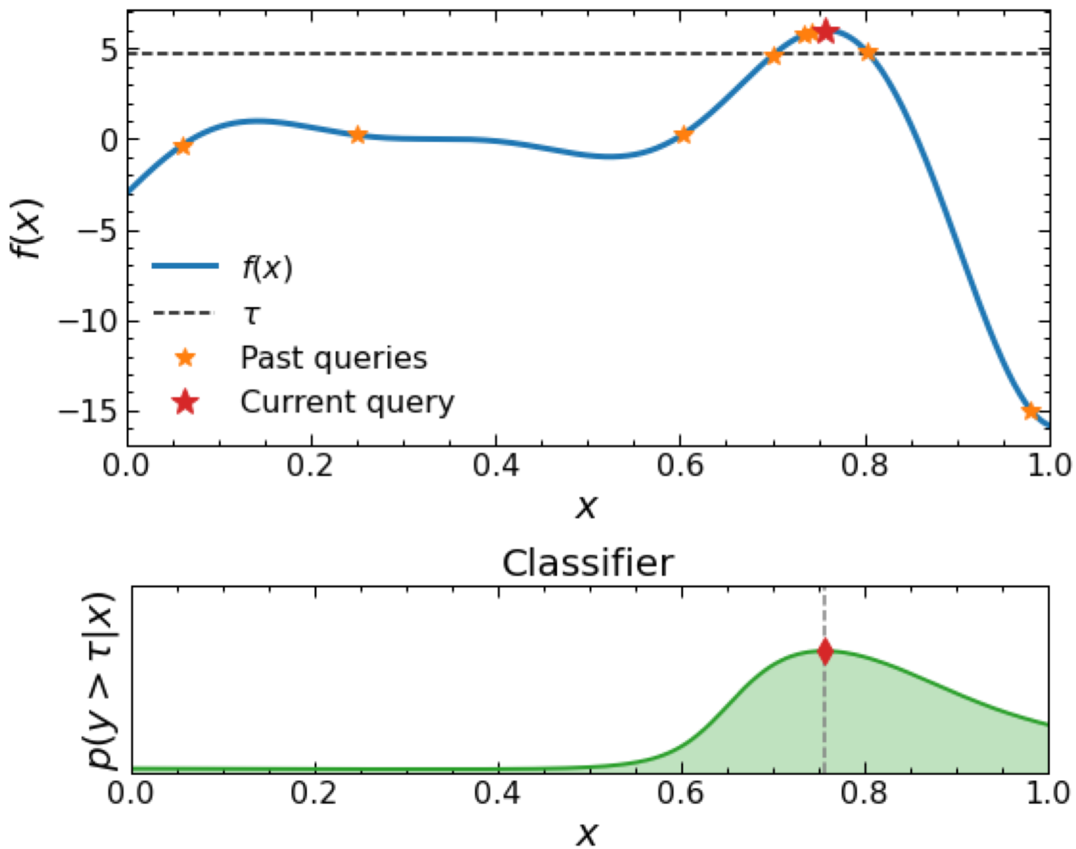
[1] **Algorithms for hyper-parameter optimization**. Bergstra, J., Bardenet, R., Bengio, Y. and Kégl, B. Advances Neural information processing systems (NeurIPS), 24. 2011

[2] **BORE: Bayesian optimization by density-ratio estimation**. Tiao, L.C., Klein, A., Seeger, M.W., Bonilla, E.V., Archambeau, C. and Ramos, F. International Conference on Machine Learning (ICML), PMLR. 2021.

Bayesian Optimization via classification

Prior work has proposed using classifiers for Bayesian optimization:

Step 5

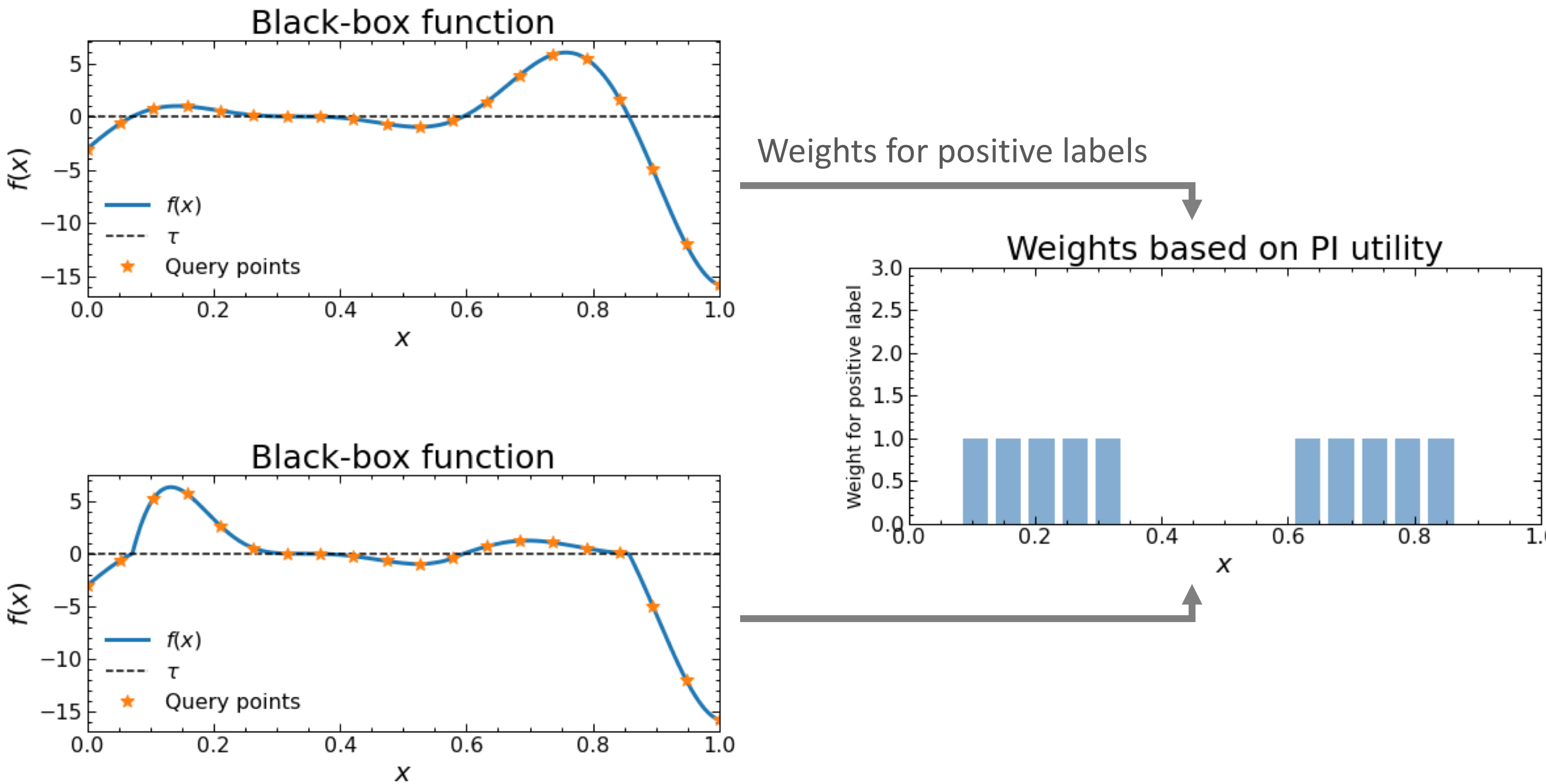


[1] **Algorithms for hyper-parameter optimization**. Bergstra, J., Bardenet, R., Bengio, Y. and Kégl, B. Advances Neural information processing systems (NeurIPS), 24. 2011

[2] **BORE: Bayesian optimization by density-ratio estimation**. Tiao, L.C., Klein, A., Seeger, M.W., Bonilla, E.V., Archambeau, C. and Ramos, F. International Conference on Machine Learning (ICML), PMLR. 2021.

Problems with current approach

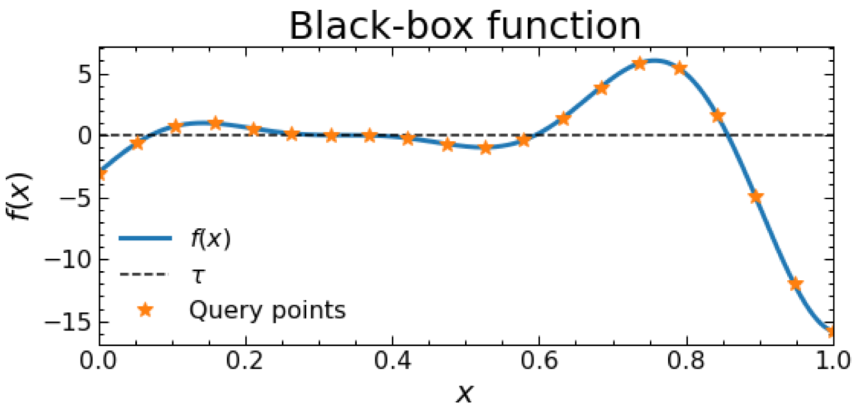
Once a point is over the threshold, it does not matter by how much

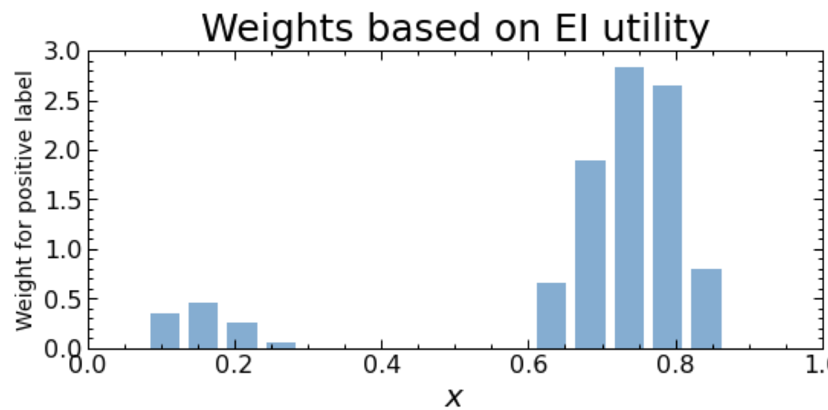


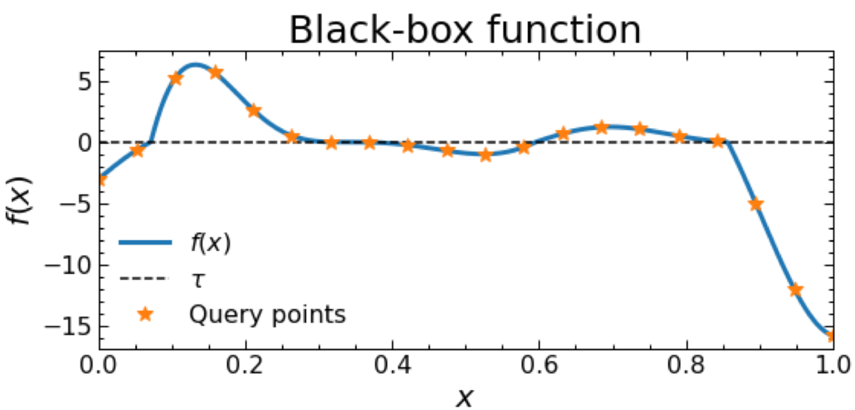
Likelihood-free Bayesian Optimization

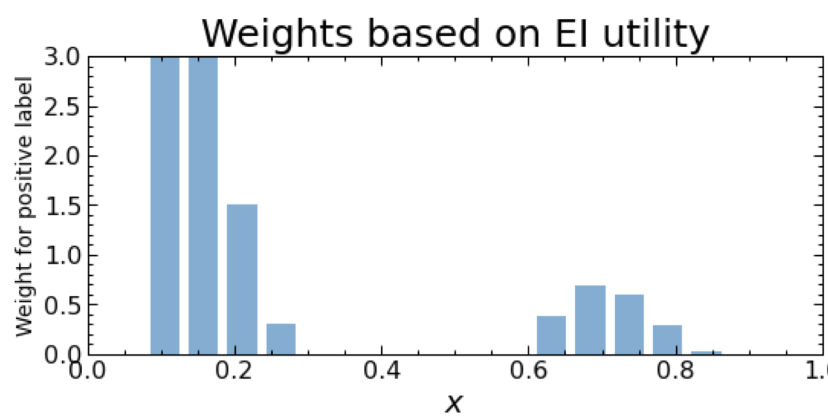
Solution: reweight the "positive" queries by its utility value.

- For EI, this becomes $$u(y) = \max(y - \tau, 0)$$
- Higher observed value leads to higher weights.









Likelihood-free Bayesian Optimization

Solution: reweight the "positive" queries by its utility value.

- In principle, works for **any non-negative utility function**!

Changes in "one line":

```
loss = nn.BCEWithLogitsLoss(weight=weight)(target, label)
```

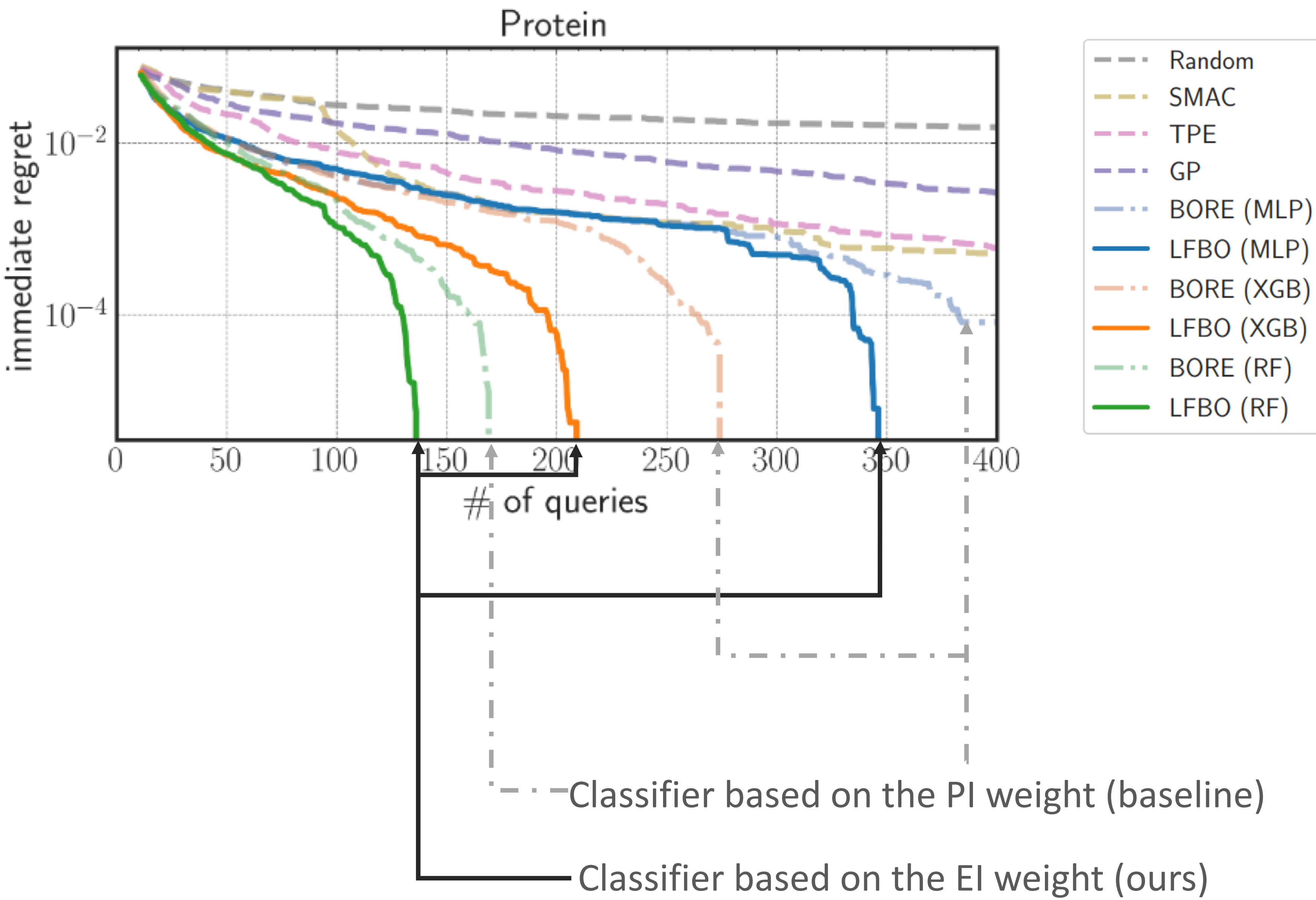
PyTorch

```
model.fit(inputs, target, sample_weight=weight)
```

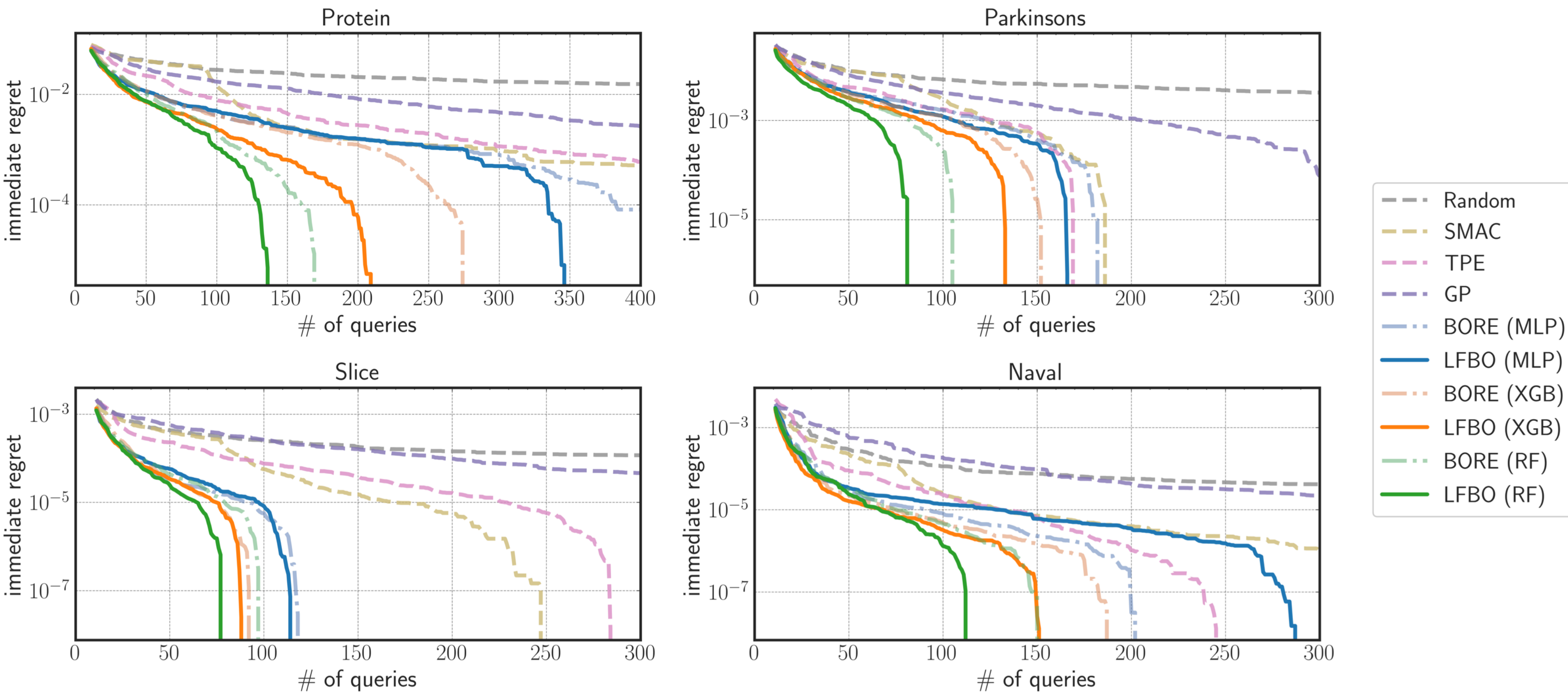
Keras, scikit-learn, XGBoost

Theory: LFBO converges to desired acquisition function asymptotically

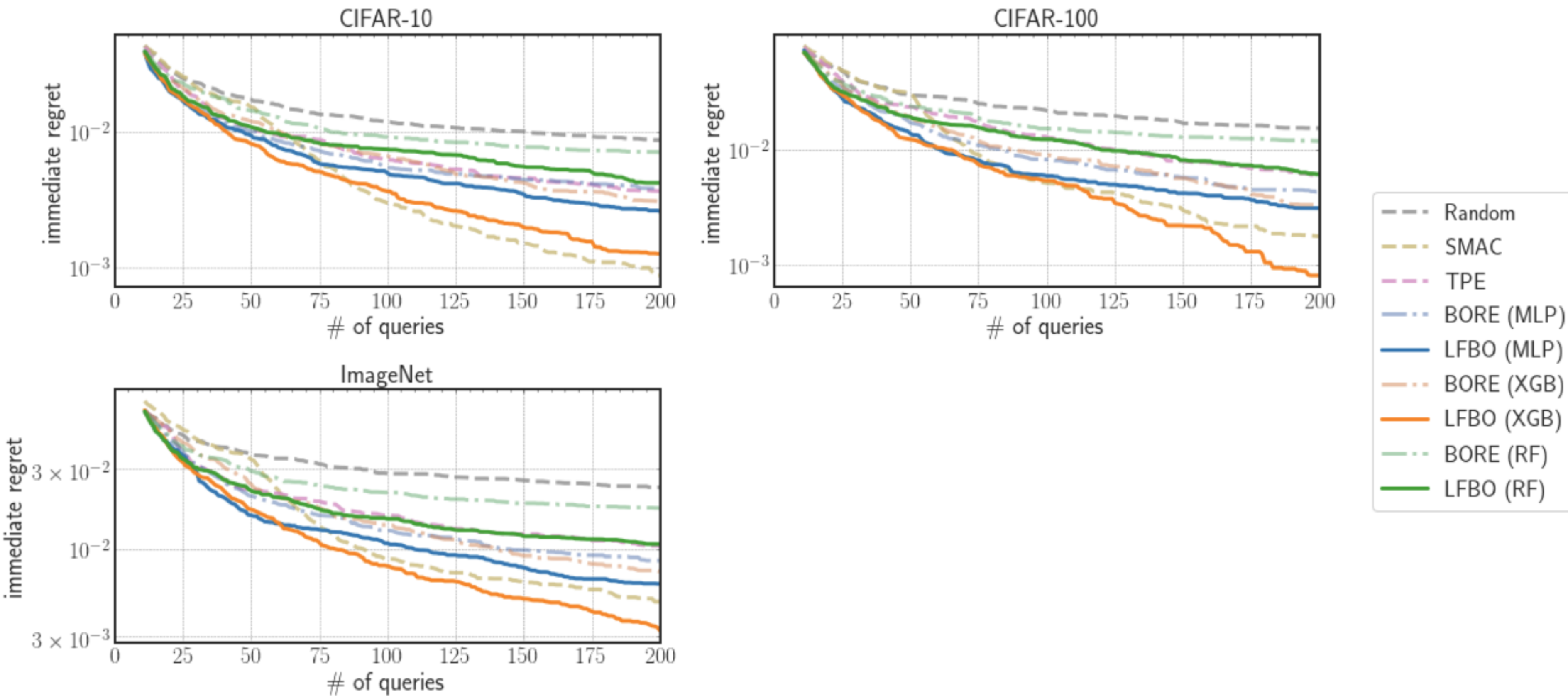
Experiments: Hyperparameter Tuning



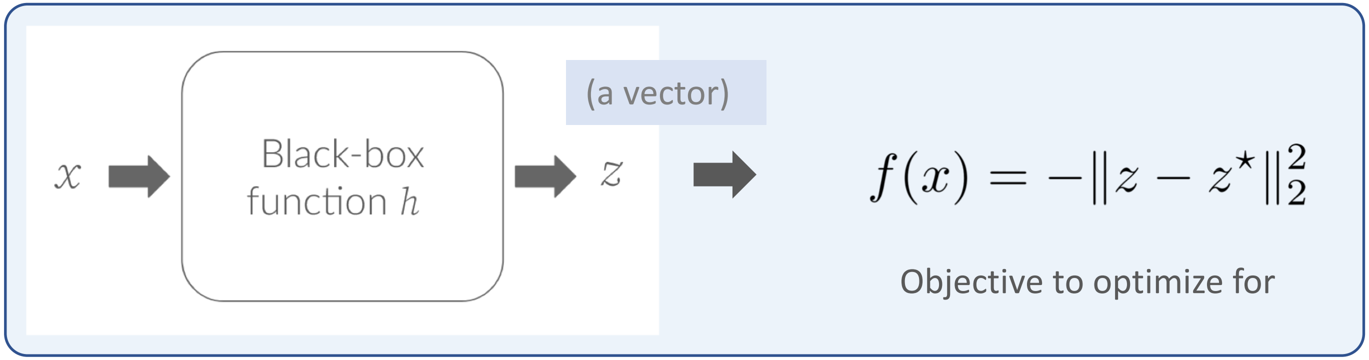
Experiments: Hyperparameter Tuning



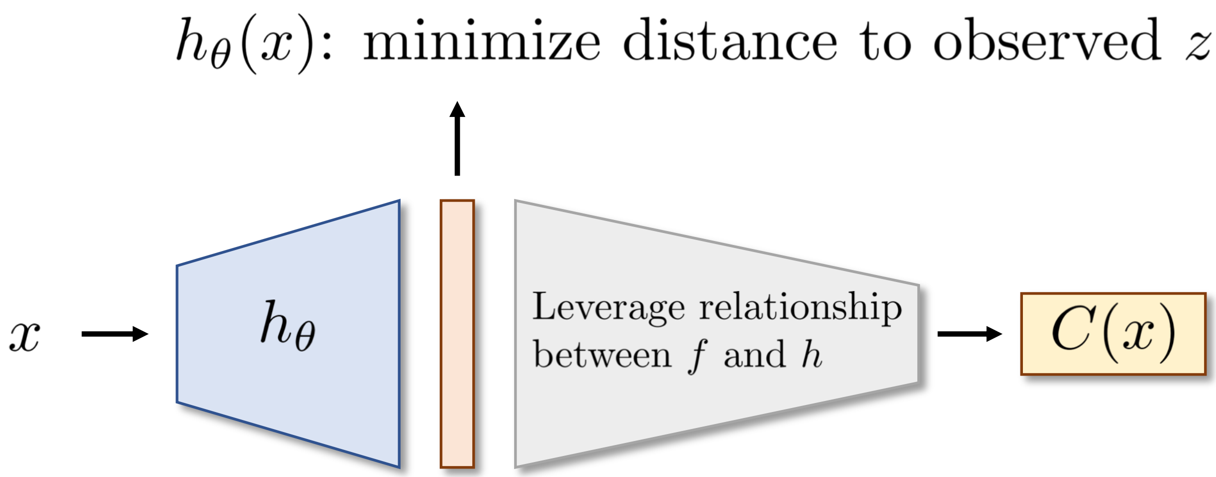
Experiments: Neural Architecture Search



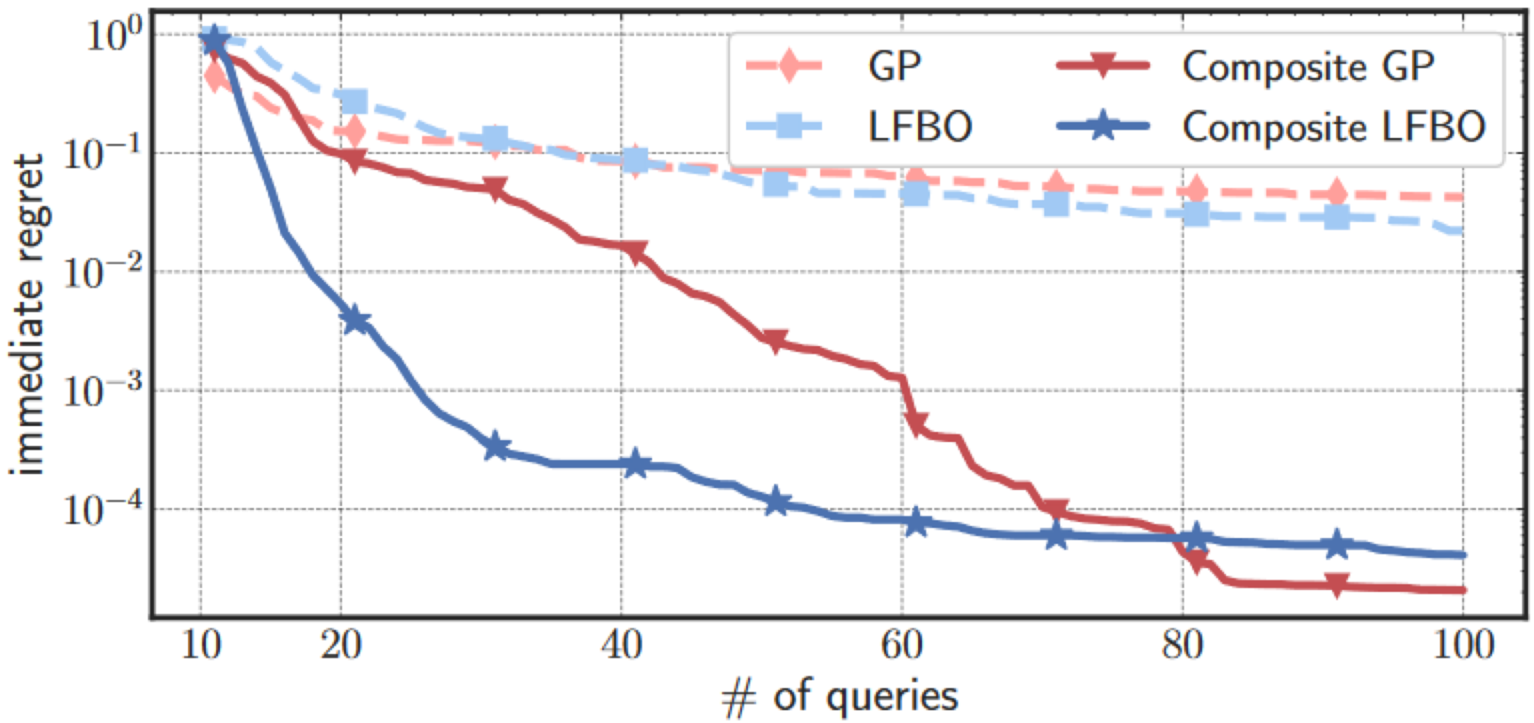
Experiments: Composite Functions



- Leveraging the structure can be helpful!
- With GPs, tractability becomes an issue.
- Easy to implement with LFBO



Experiments: Composite Functions

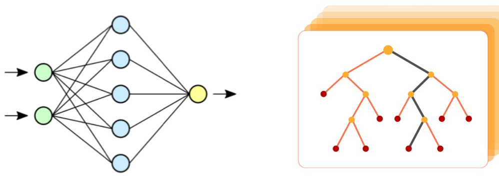


Summary

- Classifier models can be useful acquisition functions!
- To get the desired acquisition function, reweight according to utility.

LFBO (ours)

All non-negative utility functions



deterministic model

Project website: https://lfbo-ml.github.io/

Code release: https://github.com/lfbo-ml/lfbo

arXiv: https://arxiv.org/abs/2206.13035