SCHOOL OF DATA ANALYSIS

NATIONAL RESEARCH UNIVERSITY

Gaussian Processes and Bayesian Optimization

12th SHiP collaboration meeting,
2017-11-09

Artem Filatov[1,2], Alexander Baranov[1,2], Denis Derkach[1,2], Fedor Ratnikov[1,2], Oliver Lantwin[3], Andrey Ustuzhanin[1,2]

[1] Yandex School of Data Analysis [2] National Research University Higher School of Economics [3] Imperial College London

Motivation

1. In physics, we often face with optimization problems of unknown functions where gradients is intractable.
2. The computation of the target function can be very time-consuming and noisy.
3. Optimization of such function is exponentially difficult task.

Possible solution

1. We need to make different assumptions about the nature of optimizable function to simplify the task.
2. The whole optimization procedure can be decomposed into two tasks: modelling of our function and optimization of the model.

Modelling

1. We can choose any regression model to approximate the target (surrogate model).
2. But! The ability of the model to return the variance of prediction is a very useful property.
3. A variance can indicate our uncertainty about the value of the function in a certain region.
4. The most popular model with that property is a Gaussian Process regression.

Gaussian Processes

Suppose that we have the following model

$$y = \boldsymbol{w}^T \phi(\boldsymbol{x})$$

or

$$\boldsymbol{y} = \Phi \boldsymbol{w}$$

where $\boldsymbol{y}$ is target variable, $\boldsymbol{x}$ is vector of parameters, $\boldsymbol{w}$ is weights and ϕ is some mapping of original features.
Matrix Φ is a new feature function, where

$$\Phi_{ij} = \phi_j(x_i)$$

Gaussian Processes (cont'd)

Now, let introduce the prior distribution over weights.

$$p(\boldsymbol{w}) = \mathcal{N}(\boldsymbol{w}|0, \alpha^{-1}\mathbf{I})$$

We can see, that prior over weights induces probability distibution over $\boldsymbol{y}$. That distribution is normal with

$$\mathbb{E}y = m(y) = 0$$

$$var(\boldsymbol{y}) = \alpha^{-1}\Phi^T\Phi = \mathbf{K}$$

Actually, we have already constructed gaussian process (GP) with coressponding mean and covariance functions.

Gaussian Processes (cont'd). Predictive distribution.

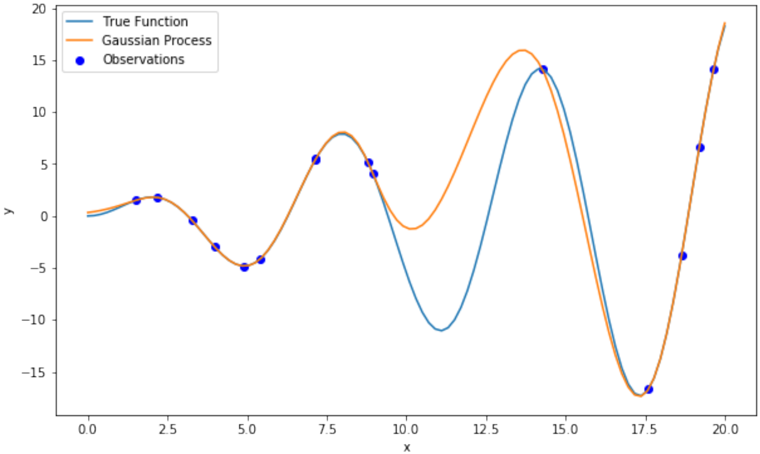
The most interesting thing for us in practice is predictive distribution.

$$p(y_{n+1}|\mathbf{X}, \boldsymbol{y}, x_{n+1}) = \frac{P(y_{n+1}, \boldsymbol{y}|x_{n+1}, \mathbf{X})}{p(\boldsymbol{y}|\mathbf{X})} = \frac{\mathcal{N}(\boldsymbol{y}_{n+1}|0, \mathbf{K}_{n+1})}{\mathcal{N}(\boldsymbol{y}_n|0, \mathbf{K}_n)}$$

With some arithmetic we can explicitly compute expectation and variance of the predictive distribution.

Gaussian Processes (cont'd). Picture.

Let look how Gaussian Process approximates $f(x) = x \sin x$ with only 14 observations sampled randomly.



Gaussian Processes (cont'd)

1. We constructed the simplest gaussian process
2. We can easily add additional assumpitons to the model: additional noise, variance structure and etc.
3. The greatest thing about GP is the predictive distribution is meaningful even far away from the data points.
4. In the same time, it's complexity is $\mathcal{O}(n^3)$
5. When $n >> 1$, we can use sparse approximations of the GP.

Bayesian Optimization

1. Let's fit a differentiable surrogate model (gaussian process) into existing data.
2. Use regular (gradient-based) optimiser to yeild the next most probable optimum point candidate.
3. The idea of Bayesian Optimization is to ask model about next point, that we should evaluate.

We will use knowledge about distribution at every point and calculate most valuable point.

Expected Improvement

What does it mean: most valuable? It depends on the task and our wishes. One approach is Expected Improvement algorithm.

EI tries to maximize.

$$\mathbb{E}[y^* - \hat{f}_n(x)]^+$$

where $\hat{f}_n(x)$ is our model constructed over n observations and y^* is the best known minima.

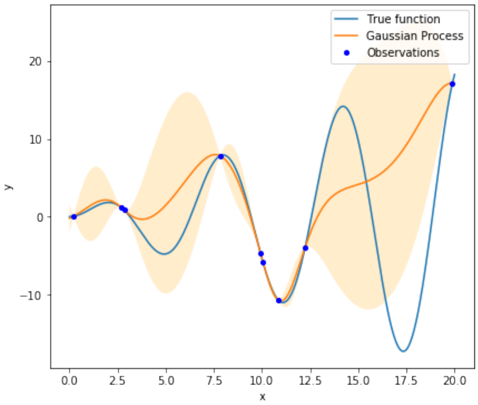
1. We would like to find a point which promises the biggest improvement of known minima.
2. EI can be computed explicitly for GP.

Full optimization cycle

The full optimization cycle will look as follows:

1. Construct surrogate model over known history.
2. Find the maxima of EI.
3. Evaluate suggested point via real physical simulation.
4. Add point to history.
5. Repeat.

Example



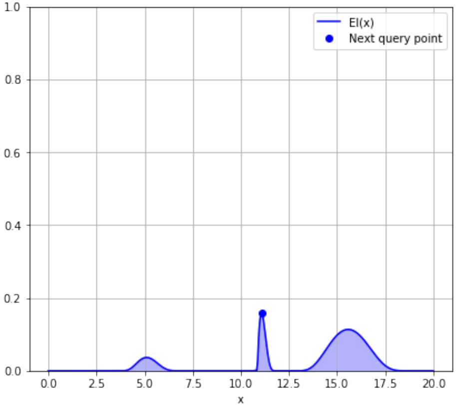


Figure: Bayesian Optimization example

Example

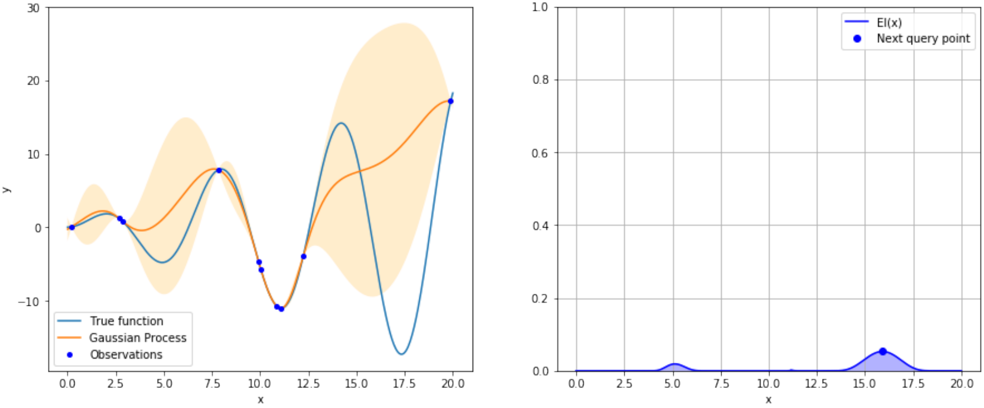


Figure: Bayesian Optimization example

Example

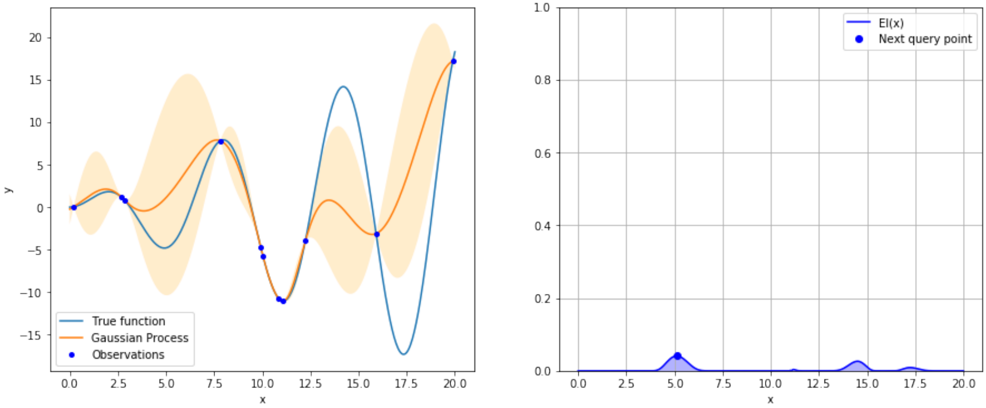


Figure: Bayesian Optimization example

Example

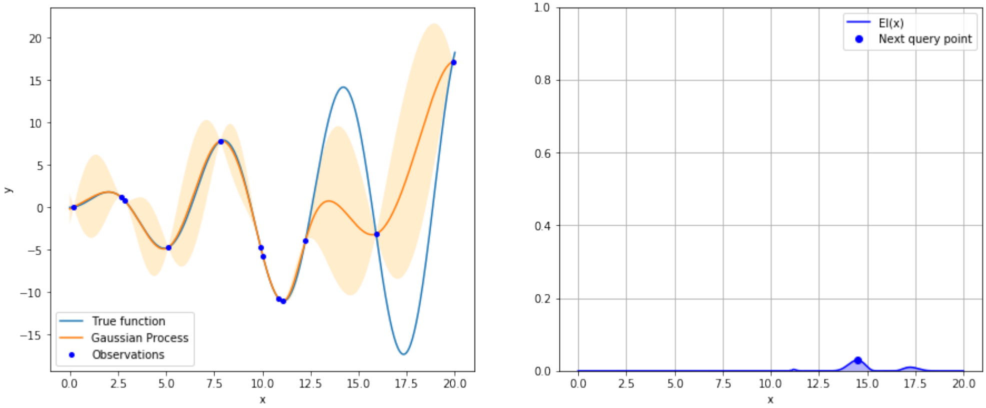


Figure: Bayesian Optimization example

Example

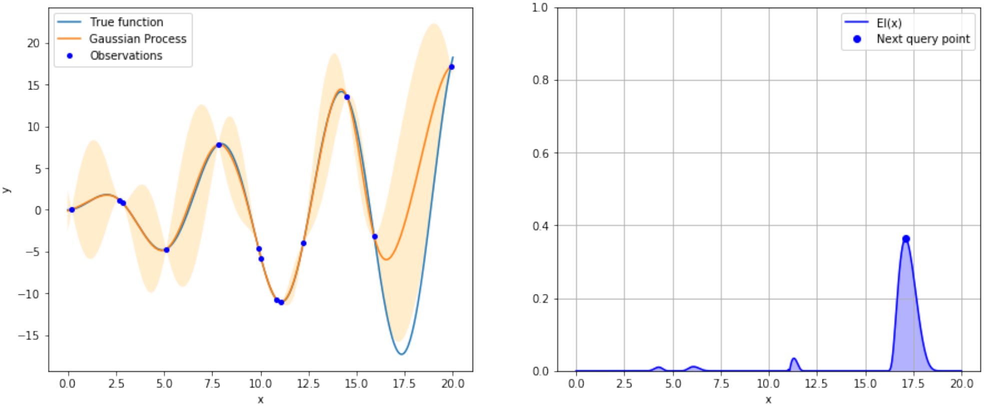


Figure: Bayesian Optimization example

Example

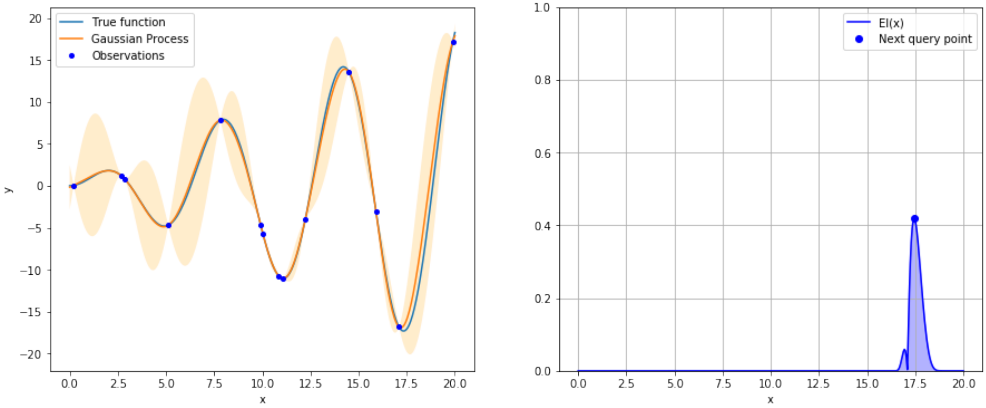


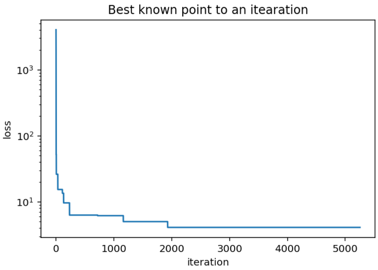
Figure: Bayesian Optimization example

Additional points about EI

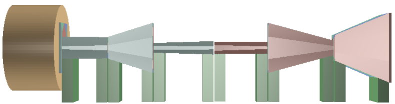
1. EI simultaneously takes into account exploration and exploitation.
2. This function is not convex, therefore we can find only approximate solution.

SHiP shield optimization

1. Bayesian Optimization was applied to optimize the muon shield.
2. We have used `scikit-optimize python3` package.
3. We have found a solution, that is lighter by 25% than baseline.



(a) Evolution of the best known point



(b) Discovered configuration

Figure: Optimization results

Conclusion

- Bayesian Optimization is a very powerful tool, which can be applied to different non-differentiable functions.
- Expected Improvement isn't the only solution. There exist various heuristical approaches.

References

› Jones et al., JoGO, 1998
› Damianou and Lawrence, AISTATS, 2013
› Bui et al., ICML, 2016
› `https://www.youtube.com/watch?v=4-pvFVd_eEQ`