Better Batch Optimizer

UNIVERSITY OF CAMBRIDGE
Department of Engineering

Y.C. Leung, Dr. M. Kok and Professor C.E. Rasmussen

Introduction

Broyden-Fletcher-Goldfarb-Shanno (BFGS) algorithm is an iterative method for solving unconstrainted non-linear multivariate optimization problems.
It involves,

1. find the search direction with the steepest gradient at a certain location.
2. perform line search to find the sub-optimal step size which can satisfy the strong Wolfe condition.
3. repeat until convergence.

Algorithm 1 BFGS [1]

Given the objective function $f(\mathbf{x})$, $\epsilon = 1 \times 10^{-4}$
Initialize starting point $\mathbf{x_0}$ and Hessian $\mathbf{H_0} = \mathbf{I}$
$k \leftarrow 0$
while $\|\nabla f_k\| > \epsilon$ **do**
  Compute search direction $\mathbf{d_k}$, $\mathbf{d_k} = -\mathbf{H_k^{-1}}\nabla f(\mathbf{x_k})$
  Compute step-size α_k via, $\alpha_k = \arg\min_\alpha f(\mathbf{x_k} + \alpha \mathbf{d_k})$
  Update $\mathbf{x_{k+1}}, \mathbf{H_{k+1}}$
  if satisfy the strong Wolfe Condition **then**
    $k \leftarrow k + 1$
  else
    break

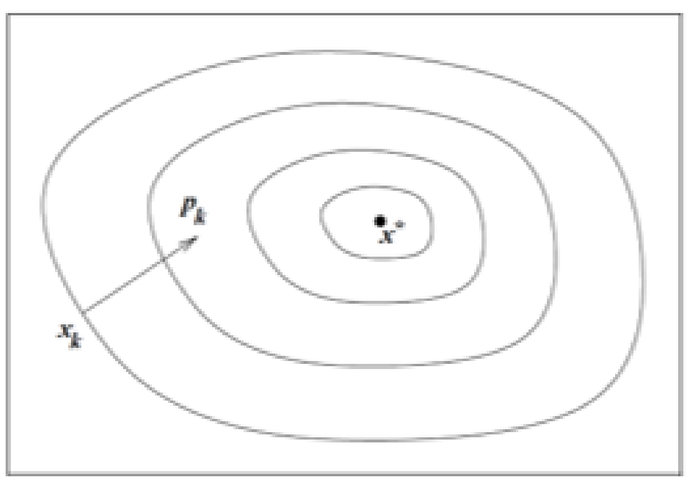

Figure: High dimensional optimization problem

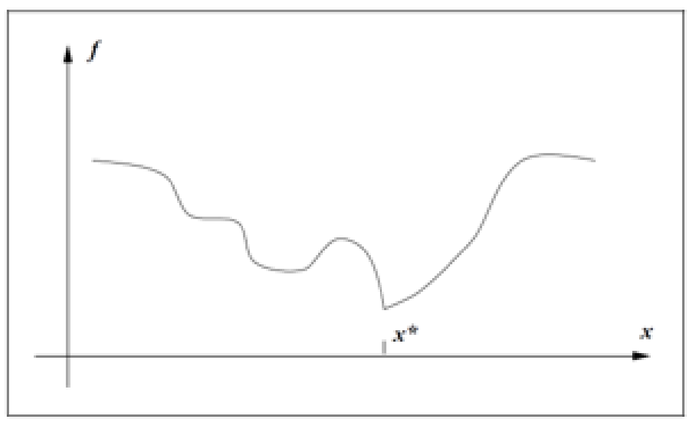

Figure: one-dimensional line search problem

Problems

There are several problems in the line search part of BFGS algorithm,

- The step-size decided cannot lead to optimal solution, but only a sub-optimal solution.
- It fails when there are noises in derivative and objective function values.
- The BFGS termination criteria (Wolfe condition: sufficient decrease condition and curvature condition) can only lead to sub-optimal solution.

Possible solutions

- Probabilistic Line Searches for Stochastic Optimization [2]

Methods

A probabilistic model is used to improve the performance of the line search algorithm. Conventionally, the Gaussian Process model is only built to describe the objective value,

$$f(\mathbf{x}) \sim \mathcal{GP}(0, k(\mathbf{x}, \mathbf{x}'))$$

The gradient $f'(\mathbf{x})$ of the objection function $f(\mathbf{x})$ can give an extra information on prediction, the Gaussian process can then be written as,

$$\begin{bmatrix} f \\ f' \end{bmatrix} \sim \mathcal{GP}\left(\mathbf{0}, \begin{bmatrix} k & k^{\partial} \\ {}^{\partial}k & {}^{\partial}k^{\partial} \end{bmatrix}\right)$$

,where ${}^{\partial^i}k^{\partial^j} = \frac{\partial^{i+j}k(x,x')}{\partial x^i \partial x^j}$. Elements of the covariance function with the cubic spline model are given by,

$$k(\mathbf{x},\mathbf{x}')=\theta^2\left(\tfrac{1}{3}\min(\mathbf{x},\mathbf{x}')^3+\tfrac{1}{2}|\mathbf{x}-\mathbf{x}'|\min(\mathbf{x},\mathbf{x}')^2\right);\ k^{\partial}=\tfrac{\theta^2}{2}((\mathbf{x}-\mathbf{x}')\min(\mathbf{x},\mathbf{x}')+\mathbf{x}\mathbf{x}')$$

$${}^{\partial}k=\tfrac{\theta^2}{2}((\mathbf{x}'-\mathbf{x})\min(\mathbf{x},\mathbf{x}')+\mathbf{x}\mathbf{x}');\ {}^{\partial}k^{\partial}=\theta^2\min(\mathbf{x},\mathbf{x}')$$

To express the prior information, the use of explicit basis functions is a way to specify a non-zero mean over functions. Consider [3],

$$g(\mathbf{x}) = f(\mathbf{x}) + \mathbf{h}(\mathbf{x})^T\beta$$

,where $f(\mathbf{x})$ is a zero mean $\mathcal{GP}$, $\mathbf{h}(\mathbf{x}) = (1, x)$ is a set of fixed basis functions and $\beta \sim \mathcal{N}(\mathbf{b}, \mathbf{B})$ are additional parameters. Therefore, we obtain another $\mathcal{GP}$,

$$g(\mathbf{x}) \sim \mathcal{GP}(\mathbf{h}(\mathbf{x})^T\mathbf{b}, k(\mathbf{x}, \mathbf{x}') + \mathbf{h}(\mathbf{x})^T\mathbf{B}\mathbf{h}(\mathbf{x}'))$$

Predictions are given by,

$$\mu\left(x^\star|y, X, \theta\right) = K_* K^{-1}\left(y - H^\top\overline{\beta}\right) + H_*^\top\overline{\beta}$$

$$\operatorname{cov}\left(x^\star|y, X, \theta\right) = K\left(x^\star, x^\star\right) - K_* K^{-1}K_*^\top + R^\top A^{-1}R$$

,where $\bar{\beta}=A^{-1}HK^{-1}y$ and $R=H_*-HK^{-1}K_*$.

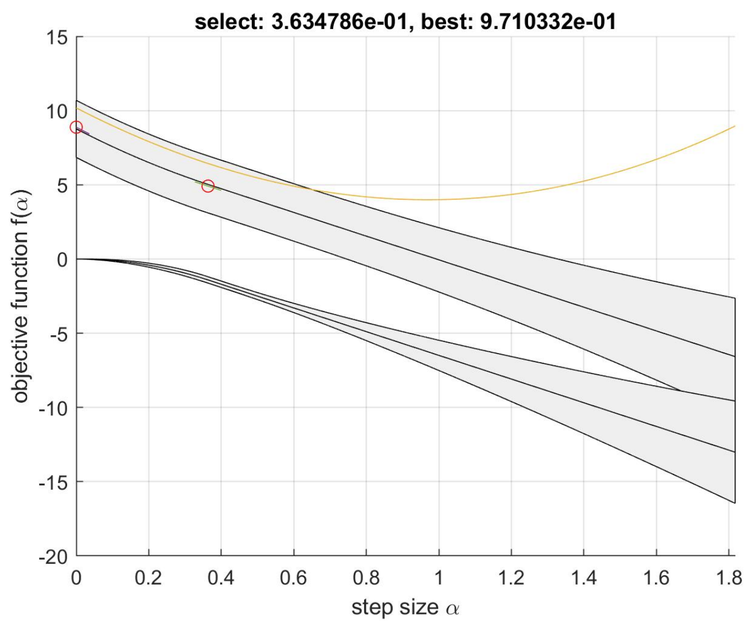

Figure: Difference between with and without mean function

To fit the model, 2 hyperparameters (noise to signal variance ratio τ^2 and signal variance θ^2) have to learn from the data. The selection of hyperparameters should lead to the minimization of negative log marginal likelihood given by,

$$-\log p(y|X) = \frac{1}{2}y^\top Z y + \frac{1}{2}\log|K| + \frac{1}{2}\log|A| + \frac{n}{2}\log 2\pi$$

The strategy is to firstly optimize τ^2 with Newton optimization method and compute θ^2 by the closed form solution.
Set $\eta = \max_{i=1,\ldots,T}\{\mu(x_i)\}$, the utility is given by [4],

$$u_{EI}(x) = \frac{\eta - \mu(x)}{2}\left(1 + \operatorname{erf}\frac{\eta - \mu(x)}{\sqrt{2\mathbb{V}(x)}}\right) + \sqrt{\frac{\mathbb{V}(x)}{2\pi}}\exp\left(-\frac{(\eta - \mu(x))}{2\mathbb{V}(x)}\right)$$

The next evaluation point is chosen as the candidate maximizing this utility. The process repeats until the optimal step-size is found.

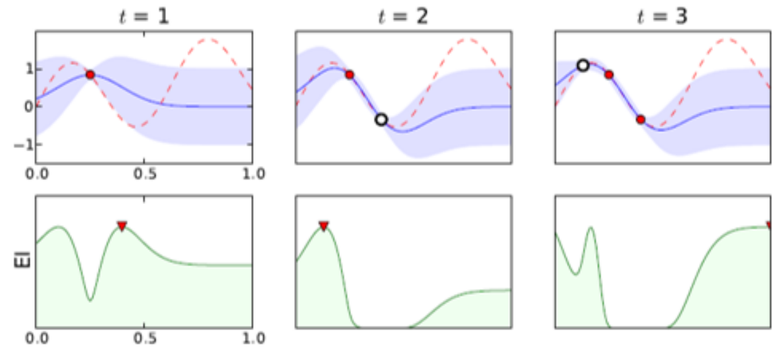

Figure: Expected Improvement. top: predictive mean and error bar. bottom: expected improvement utility.

Experiments

The prototype of the algorithm has been tested on the following 5-dimensional optimization problem,

$$f(\mathbf{x}) = 3x_1^2 + x_2^2 + 55x_3^2 + 2x_4^2 + x_5^2$$

Given some initialization parameters, it would be tested on how well it can compute the predictive mean, covariance and the next evaluation point, and find the optimal step size.

Preliminary Result

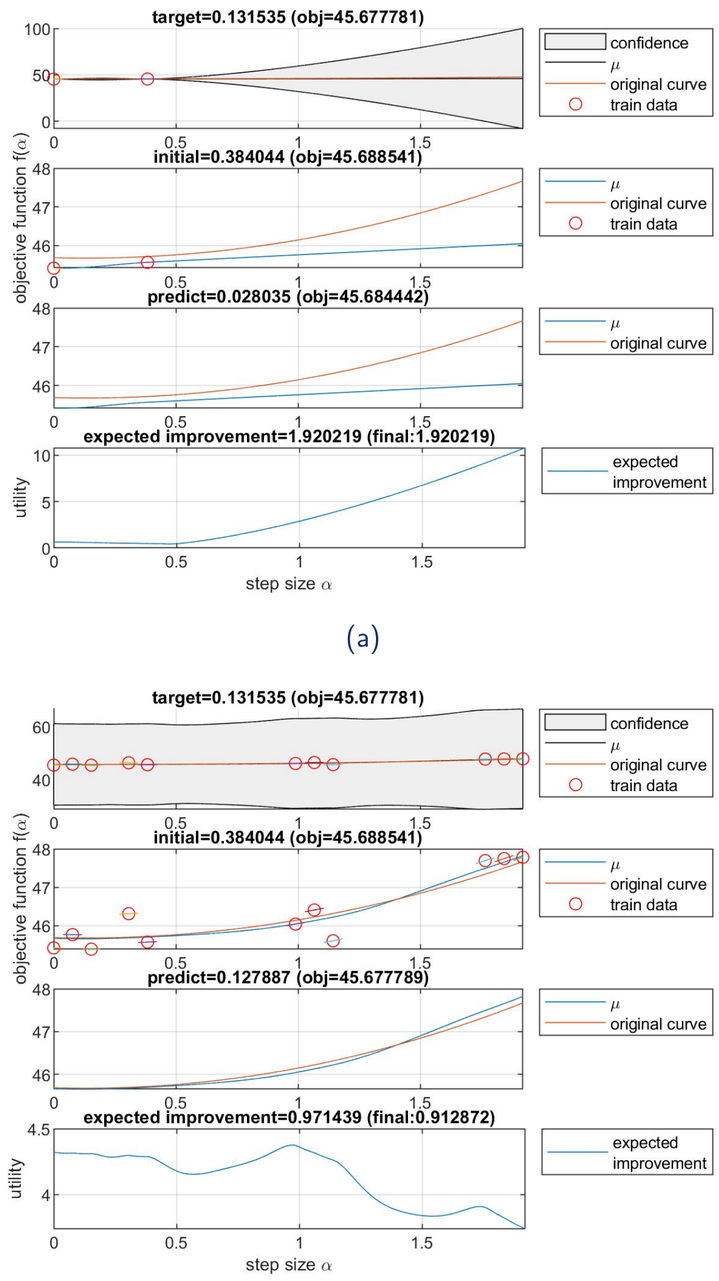

Figure: Predictions made after 1 (a) and 10 (b) iterations.

The true minimum is at 0.132 with function value of 45.678. The deterministic model has predicted 0.384 with 45.689, while the probabilistic model has predicted 0.128 with 45.678 after 10 iterations.

References

[1] J. Nocedal and S. J. Wright, *Numerical Optimization*. New York, NY, USA: Springer, second ed., 2006.

[2] M. Mahsereci and P. Hennig, "Probabilistic line searches for stochastic optimization,"

[3] C. E. Rasmussen and C. K. I. Williams, *Gaussian Processes for Machine Learning (Adaptive Computation and Machine Learning)*.

[4] E. Brochu, V. M. Cora, and N. de Freitas, "A tutorial on bayesian optimization of expensive cost functions, with application to active user modeling and hierarchical reinforcement learning,"