

Übungen zur Vorlesung

Praktische Optimierung, SoSe 2022

Prof. Dr. Günter Rudolph, Dr. Roman Kalkreuth

<https://ls11-www.cs.tu-dortmund.de/people/rudolph/teaching/lectures/POKS/SS2022/lecture.jsp>**Blatt 9, Block B**

08.06.2022

Abgabe: 15.06.2022**Aufgabe 9.1: Modellbasierte Optimierung (10 Punkte)**

In dieser Aufgabe beschäftigen wir uns mit der modellbasierten Optimierung (MBO). MBO wird verwendet, wenn die Zielfunktionsauswertungen teuer sind, und somit kleine Budgets an Zielfunktionsauswertungen zur Verfügung stehen. Im Rahmen der Übung wird jedoch darauf verzichtet Funktionen zu verwenden, die viel Zeit zur Auswertung benötigen wie beispielsweise aufwändige Simulationen, die mehrere Stunden oder Tage benötigen, da die Lösung der Aufgabe nicht überwiegend daraus bestehen soll auf die Funktionsauswertungen zu warten. Es werden jedoch nur wenige Zielfunktionsauswertungen zugelassen.

Beschreiben Sie zunächst das Vorgehen bei der modellbasierten Optimierung (EGO-Algorithmus) in eigenen Worten. Dazu können Sie auch nochmal das Vorgehen in *Efficient Global Optimization of Expensive Black-Box Functions* von Jones et al. in *Journal of Global Optimization*, volume 13, pages 455–492 (1998) nachlesen.

Sie haben nun die Aufgabe mithilfe von MBO die folgenden beiden Funktionen zu minimieren:

- (a) $f(x) = x^2 - 10 \cos(5x)$, $x \in [-10, 10]$,
- (b) $f(x, y) = x^2(1 + \cos(y)) + y^2 \sin(x)$, $x, y \in [-10, 10]$.

Dazu benötigen Sie zunächst ein initiales Design. Hierzu sollen Sie ein Improved Latin Hypercube Design (Paket `lhs`) mit 5 initialen Punkten für Funktion (a) und 10 initialen Punkten für Funktion (b) verwenden.

Nutzen Sie für MBO die Funktion `mbo` aus dem Paket `mlrMBO`. Der folgende Link könnte Ihnen dabei weiterhelfen: <https://mlr-org.github.io/mlrMBO/>.

Analysieren Sie, ob als Infill-Kriterium Expected Improvement (EI) oder (Lower) Confidence Bound (LCB) zu besseren Ergebnissen bei der Optimierung führt. Gehen Sie hierbei nicht nur auf den optimalen Wert sondern auch auf die Abdeckung des Suchraums ein.

Stellen Sie Ihre Ergebnisse grafisch dar. In den Grafiken sollten einerseits die Zielfunktionen in dem zulässigen Bereich dargestellt sein und andererseits alle Schritte der Optimierung. Dabei soll jede Iteration mit einer Zahl versehen sein, wobei die Punkte aus dem initialen Design eine 0 erhalten sollen und die nachfolgenden Punkte die Nummer der jeweiligen Iteration. Die Nummern sollen sich möglichst nicht überdecken. Heben Sie den besten gefundenen Wert farbig hervor. Interpretieren Sie Ihre Grafiken.

Um besser zu verstehen wie die beiden verwendeten Infill-Kriterien funktionieren, sollen Sie jeweils angeben, wie sich EI und LCB aus den Ergebnissen von MBO berechnen lassen. Sie sollen also die Werte für EI und LCB nachrechnen und überprüfen, dass Sie zu den gleichen Ergebnissen kommen.

Zu beachten:

- Nutzen Sie für MBO mit Expected Improvement und (Lower) Confidence Bound das gleiche initiale Design. Die initialen Designs für Funktion (a) und (b) sollen sich hingegen unterscheiden.
- Nutzen Sie als Metamodell Kriging (Paket `DiceKriging`).
- Verwenden Sie 10 MBO-Iterationen für Funktion (a) und 15 MBO-Iterationen für Funktion (b).
- Setzen Sie für Ihre Abgabe in der Funktion `mbo` den Parameter `show.info = FALSE`.
- Die Parameter `filter.proposed.points` bzw. `filter.proposed.points.tol` können bei der Behebung numerischer Probleme helfen.
- Die Werte für EI und LCB lassen sich nur für die Punkte berechnen, die nicht zum initialen Design gehören. Bei EI erhalten Sie mit -1 multiplizierte Ergebnisse. Überlegen Sie sich wieso es sinnvoll sein könnte, dass sich für EI Ergebnisse in dieser Form ergeben. Schreiben Sie Ihre Vermutung auf.