

ユーザの選好推定に基づく 多目的ベイズ最適化による ハイパーパラメータ最適化

平井 遥斗

富山県立大学 情報システム工学科

2023 年 7 月 14 日

はじめに

多目的最適化では、目的関数間のトレードオフにより、一般には解は一意に定まらない。

実問題においてユーザが多目的最適化を利用する際には、列挙された解の中から必要な解の一つを選んで採用することが多い。

そこで、ユーザが解を選択するのではなく、多目的最適化の段階でユーザがどのような解を必要としているのかという情報を考慮することで、ユーザが求める解のみを効率的に発見したい。

本研究の目的

本研究では、多目的最適化に関するユーザの好みを、ユーザが具体的な値として直接与えるのではなく、より容易に与えられる情報として、「解の良さの比較」や「目的関数値の改善要望から」逐次的にベイズ推定し、それを多目的最適化に取り入れることを考える。

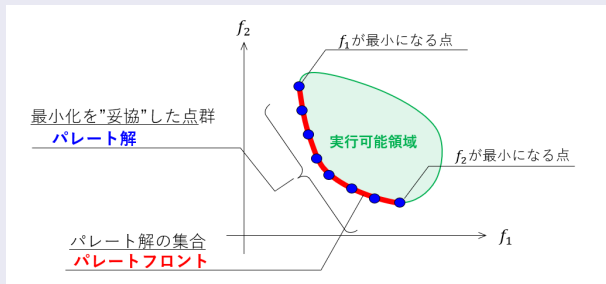
多目的最適化とパレート最適性

3/10

多目的最適化

複数の目的関数を同時に最適化する問題は多目的最適化と呼ばれる。

通常多目的最適化の目的関数間にはトレードオフの関係があり、最適解が一意には定まらず、パレート最適と呼ばれる性質を満たす解が複数存在し得る。



多目的最適化ではこのパレート解の列挙を目指すことが最も一般的である。

問題設定

問題によってはパレート解は大量に存在し、ユーザが採用したい解を一つ選択することになる場合が多い。

そこで、多目的最適化にユーザの好みを導入することで、ユーザの求める解を効率的に発見することを考える。

目的関数 $f^l(x)$ について、ユーザがどの目的関数をどの程度重視しているかを考慮した最大化問題を考える。

ある解 f_x についてのユーザにとっての好ましさを表す関数 U_ω が存在するとする。

ユーザの選好最適化は U_ω の x に関する最大化として解釈できる。

$$x_* := \operatorname{argmax} U_\omega(f_x) \quad (1)$$

選好推定を用いた多目的ベイズ最適化

5/10

提案フレームワークの構成

直接与えることが困難であるユーザの選好を推定し、それを用いた多目的最適化を提案する.

提案手法は3つの部分から構成される.

- (1) 目的関数の推定.
- (2) ユーザへの選好に関する情報として、ユーザがある2つの解についてどちらを好むのかという情報と、ある会においてユーザがどの目的関数値を改善したいかという情報の2種類を与える.
- (3) 選好の推定と、選好を用いた獲得関数の計算.
ユーザから得られる選好に関する情報から選好をベイズ的に推定し、ベイズ最適化における獲得関数の計算に用いる.

選好表現

ユーザの選好を表現する手法として、解の良さを連続的に評価できる、ベクトル表現に基づく効用関数を考える。

まず、選好を表すベクトルを $\omega = [w_1, \dots, w_L]^\top$ と定義し、制約を $\omega_i > 0, \|\omega\|_1 = 1$ とする。

効用関数は、多目的最適化のある入力 x に対する f_x を入力にとり、その x に対する好ましさの度合いをスカラー値として返すものとし、以下のように定義する。

$$U_\omega(f_x) = \min\left(\frac{f_x^1}{\omega_1}, \dots, \frac{f_x^L}{\omega_L}\right) \quad (2)$$

以上のように、効用関数を定義することで、多目的最適化の解に対するユーザの選好を表現することができた。

選好のベイズ推定

ユーザから得られる相対的な選好情報からパラメータ ω をベイズ的に推定することを考える。

事前分布として、 ω の制約を求めたことから、この制約を満たす確率分布である Dirichlet 分布を採用する。

$$p(\omega; \alpha) = \frac{1}{B(\alpha)} \prod_{i=1}^L \omega_i^{\alpha_i - 1} \quad (3)$$

α は集中パラメータ、 B はベータ関数である。

ユーザが与えられる選好表現を考え、この時 ω の事前分布がどのように得られるかを示す。

まず一対比較として解 f_{x_i} が解 $f_{x'_i}$ よりも良い解であると提示した場合を考える。これを $f_{x_i} \succ f_{x'_i}$ と表す。

一対比較はこれにガウスノイズを加えた後の大小関係として比較され则认为ると。

$$f_{x_i} \succ f_{x'_i} \Leftrightarrow U_{\omega}(f_{x_i}) + \epsilon_i > U_{\omega}(f_{x'_i}) \quad (4)$$

のように表現できる。

選好のベイズ推定

n 個の比較 $\{f_{x_i} \succ f_{x'_i}\}_{i=1}^n$ が得られた時, ω の尤度は以下のようになる.

$$L(\omega | \{f_{x_i} \succ f_{x'_i}\}_{i=1}^n) = \prod_{i=1}^n \Phi\left(\frac{U_\omega(f_{x_i}) - U_\omega(f_{x'_i})}{\sqrt{2}\alpha}\right) \quad (5)$$

Φ は標準正規分布の累積分布関数である.

次に, ユーザがある解 f_{x_j} について, 目的関数 $f_{x_j}^{l_j}$ を目的関数 $f_{x_j}^{l'_j}$ より改善したい場合を考える. これを $f_{l_j} \succ f_{l'_j}$ と表す.

一対比較の場合と同じ考え方により, 尤度は以下のように表現できる.

$$L(\omega | \{l_j \succ l'_j\}_{j=1}^m) = \prod_{j=1}^m \Phi\left(\frac{g^{l_j}(f_{x_j}) - g^{l'_j}(f_{x_j})}{\sqrt{2}\alpha}\right) \quad (6)$$

計算機実験

計算機実験により、提案手法の評価を行う。

人工データ実験による評価を行い、実験結果を示す。

評価指数は真の選好パラメータ ω_{true} を用いた simple regret とし、この simple regret が小さいほどユーザにとって良い解を発見できているということになる。

提案手法に対する比較方法は、候補点の中から一様ランダムに次の探索点を決める Random と、選好を用いない通常の多目的最適化 (MOBO-RS) とする。

目的関数として、多目的最適化のベンチマーク関数として使われる DTLZ1 を用いる。

実験結果

図から、提案手法が最も速く simple regret の値が小さくなっていることがわかる。

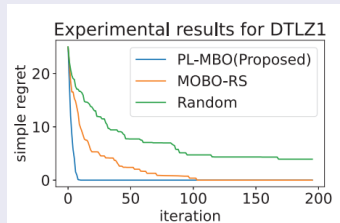


図 1: ベンチマーク関数を用いた Simple regret の推移比較

つまり、あらかじめ仮定したユーザの真の選好に沿った好ましい解が、提案手法によって効率的に探索できていることがわかる。

まとめ

本論文では多目的ベイズ最適化にユーザの好みの推定を導入し、ユーザにとって望ましい解を効率的に探索するフレームワークを提案した。また、計算機実験により、提案手法が有効であると示した。