

機械学習におけるハイパパラメータ 最適化手法

尾崎 嘉彦 野村 将寛 大西 正輝

安藤 祐斗

富山県立大学 電子情報工学科

November 30, 2020

はじめに

ハイパパラメータ
最適化

グリッドサーチ

ランダムサーチ

ベイズ最適化

Nelder-Mead 法

グレーボックス最
適化

最適化手法選択

まとめ

背景

近年、機械学習モデルの性能は改善の一途を辿り、一般物体認識をはじめとした画像認識, ゲーム, AI などの幅広い分野で活用されている. また, 性能改善と共にモデル構造が複雑化され, チューニングや設計は高度な知識と経験が要求されるようになった.

目的

一般に, 機械学習モデルの性能を十分に引き出すためには, 機械学習の挙動を設定するパラメータである, ハイパパラメータを適切に設定する必要がある. 本スライドでは, 最適化アルゴリズムの用いて機械学習モデルをチューニングするハイパパラメータ最適化手法について紹介する.

ハイパパラメータ最適化について (1)

3/18

ハイパパラメータ最適化問題の定式化

機械学習モデルの持つチューニング可能な n 個のハイパパラメータの定義域を $X_i (i = 1, \dots, n)$ とする. ハイパパラメータ設定の探索空間 $\mathbf{X} = X_1 \times \dots \times X_n$ は様々な種類のハイパパラメータを含み, $f: \mathbf{X} \rightarrow \mathbb{R}$ をモデルの性能を示す損失関数とする. このとき, ハイパパラメータ最適化は, モデルが最良性能を達成するハイパパラメータの設定を見つけ出す. 次のような最小化問題として定式化できる.

$$\text{minimize } f(x)$$

$$x \in \mathbf{X}$$

はじめに

ハイパパラメータ最適化

グリッドサーチ

ランダムサーチ

ベイズ最適化

Nelder-Mead 法

グレーボックス最適化

最適化手法選択

まとめ

ハイパパラメータ最適化について (2)

4/18

はじめに

ハイパパラメータ
最適化

グリッドサーチ

ランダムサーチ

ベイズ最適化

Nelder-Mead 法

グレーボックス最
適化

最適化手法選択

まとめ

例として, 一般物体認識を行う深層ニューラルネットワークのチューニングとすると, 具体的に下のように記述される.

minimize $f(x; \omega^*, D_{valid})$

subject to $\omega^* = \operatorname{argmin}_{\omega} g(x, \omega; D_{train})$

$x \in X$

D_{valid} : 検証データセット

f : モデルの性能を示す損失関数

ω : モデルの重み

D_{train} : 学習データセット

g : 学習に用いる損失関数

ハイパパラメータ最適化について (3)

5/18

ハイパパラメータ最適化問題の持つ性質

- ・ハイパパラメータ最適化の目的関数（機械学習の性能を示す損失関数）の評価コストが高い.
- ・ハイパパラメータ最適化問題は典型的には数十次元程度までの探索空間を扱う.
- ・機械学習モデルの持つハイパパラメータ全体のうち, 性能に関して重要なものがごく一部しかない性質→ Low Effective Dimensionality(LED), これに強いことが望まれる.
- ・目的関数評価の不確実性を扱える.

はじめに

ハイパパラメータ最適化

グリッドサーチ

ランダムサーチ

ベイズ最適化

Nelder-Mead 法

グレーボックス最適化

最適化手法選択

まとめ

グリッドサーチとは

- ・各ハイパパラメータに対して幾つかの代表値を選択し、ハイパパラメータ設定の探索空間をその直積としその空間を全探索する.
- ・ブラックボックス最適化問題 (入力 λ に対して、出力 $L(\lambda)$ は得られるが、最小化したい目的関数 $L(\lambda)$ の勾配 $\nabla L(\lambda)$ 等の情報は手に入らない) に扱われる手法

メリット

- ・人間にとって直感的
- ・選択した代表値によって評価されるハイパパラメータ設定が確定するため、すべての目的関数評価を非同期に並列化できる.

デメリット

- ・LED を持った探索に弱い.
- ・探索の途中で目的関数評価から得られる情報は一切活用できない.

最適化例について

図 2 に示す関数 $f(x, y) = (x - \frac{3}{4})^2 + \frac{y}{100}$ は探索空間 $[0, 1]^2$ において LED を持ち, 変数 x が目的関数値に対して支配的である. よって x の値が異なる評価だけが探索に役立つ. グリッドサーチを行った結果, 図 2 の 9 回の評価のうち, 6 回は x の値が変化していないため無駄な評価である, よってグリッドサーチは LED に弱いといえる.

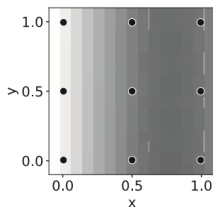


図 1: グリッドサーチを用いた LED を持つ関数 $f(x, y) = (x - \frac{3}{4})^2 + \frac{y}{100}$ を最適化した例

ランダムサーチとは

- ・乱数生成器を用いてハイパパラメータ設定を生成する.
- ・ブラックボックス最適化問題に扱われる手法.

メリット

- ・全ての評価を非同期的に並列化が可能である
- ・グリッドサーチと比較して LED に強い.

デメリット

- ・探索の途中で目的関数評価から得られる情報は一切活用できない.

最適化例について

図 2 に示す関数 $f(x, y) = (x - \frac{3}{4})^2 + \frac{y}{100}$ は探索空間 $[0, 1]^2$ において LED を持ち, 変数 x が目的関数値に対して支配的である. よって x の値が異なる評価だけが探索に役立つ. ランダムサーチを行った結果, 図 2 の 9 回の評価全てにおいて x が変化しているため, 無駄な評価が生じない.

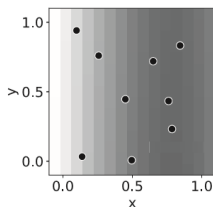


図 2: ランダムサーチを用いた LED を持つ関数 $f(x, y) = (x - \frac{3}{4})^2 + \frac{y}{100}$ を最適化した例

ベイズ最適化とは

- ・最も成功を収めているブラックボックス最適化手法
- ・代理モデルと獲得関数の二つの要素からなる.

代理モデルとは

- ・ハイパパラメータ設定と目的関数値のペアデータから評価コストの高い目的関数や有望なハイパパラメータ設定の分布を近似するために用いられる
- ・代表的な代理モデル：ガウス過程（連続関数に対する確率過程であり, 出力値の平均を定める平均関数と, 2 点間の距離を定めるカーネルによって定義される）

獲得関数とは

- ・ 効率的に有望なハイパパラメータをサンプルするための指標
- ・ 代表的な獲得関数：EI(Expected Improvement, あるしきい値 (既知の最良の目的関数値) に対する, ハイパパラメータ設定 x における $f(x)$ の改善量の期待値を表す指標)

ベイズ最適化手順

- 1 観測データから代理モデルを構築, 更新する.
- 2 構築した代理モデルから獲得関数を計算し, その獲得関数を最大化することで次に評価するハイパパラエータ設定を選択する.
- 3 選択したハイパパラメータ設定を評価する.

はじめに

ハイパパラメータ
最適化

グリッドサーチ

ランダムサーチ

ベイズ最適化

Nelder-Mead 法

グレーボックス最
適化

最適化手法選択

まとめ

Nelder-Mead 法とは

- $m + 1$ 個の頂点からなる m 次元の単体をアメーバのように動かしながら関数の最小値を反射, 膨張, 収縮の 3 種類を使い分けながら探索するアルゴリズム
- SVM や畳込みニューラルネットワークのチューニングにおいて実績がある.

メリット

- 局所的な探索を行うため, 良いハイパパラメータ設定を見つけられたとき, その近傍のより良いハイパパラメータ設定を積極的に探索する能力をもつ

デメリット

- 局所的な探索は初期値への依存が高く, 悪質な局所解に陥る可能性がある.

具体的な手法 (反射)

- 1 各頂点の座標を $p_i (i = 1, 2, \dots, m + 1)$ とする. 頂点の初期位置を決める
- 2 頂点を序列化する.
(例) $f(\vec{p_1}) \leq f(\vec{p_2}) \leq \dots \leq f(\vec{p_m}) \leq f(\vec{p_{m+1}})$
最良点 ($f(\vec{p_1})$), 第二最悪点 ($f(\vec{p_m})$), 最悪点 ($f(\vec{p_{m+1}})$)
- 3 重心 $\vec{p_0} = \frac{1}{m} \sum_{i=1}^m \vec{p_i}$ を求める.
- 4 最悪点 ($\vec{p_{m+1}}$) を重心に対して反射点 $\vec{p_{ref}}$ を求める.
$$\vec{p_{ref}} = \vec{p_0} + (\vec{p_0} - \vec{p_{m+1}})$$

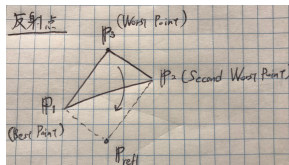


図 3: 反射の様子

具体的な手法 (膨張)

- 5 反射点 $\overrightarrow{p_{ref}}$ が今までの最良点よりは悪いが, 第二最悪点より良い場合, $(f(\overrightarrow{p_1}) < f(\overrightarrow{p_{ref}}) < f(\overrightarrow{p_m}))$ 今までの最悪点は捨て, 反射点に入れ替える. 今までの第二最悪点が新しい最悪点となる.
- 6 $f(\overrightarrow{p_{ref}}) < f(\overrightarrow{p_1})$ の場合, 反射拡大点 $\overrightarrow{p_{expand}}$ 求める.

$$\overrightarrow{p_{expand}} = \overrightarrow{p_0} + 2(\overrightarrow{p_0} - \overrightarrow{p_{m+1}}) = 3\overrightarrow{p_0} - 2\overrightarrow{p_{m+1}}$$
 - 1 $f(\overrightarrow{p_{expand}}) < f(\overrightarrow{p_{ref}}) < f(\overrightarrow{p_1})$ の場合,
最悪点と反射点は捨て, 反射拡大点を新しいグループの最良点 ($\overrightarrow{p_1}$) として採用する.
 - 2 $f(\overrightarrow{p_{ref}}) \leq f(\overrightarrow{p_{expand}})$ の場合,
最悪点と反射拡大点を捨て, 反射点を新しいグループの最良点 ($\overrightarrow{p_1}$) として採用する.

はじめに

ハイパラメータ
最適化

グリッドサーチ

ランダムサーチ

ベイズ最適化

Nelder-Mead 法

グレーボックス最
適化

最適化手法選択

まとめ

図 4: 収縮の様子

グレーボックス最適化とは

- ・ 目的関数値に加えて、対象問題の特徴から得られる最適化に有益な補助情報を活用し、従来のブラックボックス最適化手法を高速化する.

学習の早期打ち切り

- ・ 機械学習モデルの、反復的に損失関数を最小化して学習する経過を監視し、現在のハイパパラメータ設定における学習を続けても他の設定と比べて性能が勝る見込みがないと思われる場合、早期に学習を停止する.

ウォームスタート

- ・ 過去に解いたハイパパラメータ最適化問題の結果を利用することで、従来のブラックボックス最適化手法の初期化を改良し、最適化を高速化する.

最適化手法の選択

以下の観点から, 適切なハイパパラメータ最適化を行う上で最適化手法を選択する.

- ・ 逐次評価回数の上限值: チューニングに費やせる時間と機械学習モデルの学習にかかる時間の比により定まる.
- ・ 並列計算リソース: 費やせる予算や作業環境などに依存して定まる.
- ・ ハイパパラメータの種類: モデルのどのハイパパラメータをチューニング対象とするか選択することで定まる.

はじめに

ハイパパラメータ最適化

グリッドサーチ

ランダムサーチ

ベイズ最適化

Nelder-Mead 法

グレーボックス最適化

最適化手法選択

まとめ

まとめ

- ・様々なハイパパラメータ最適化手法があり, それぞれにメリットとデメリットが存在するため, 状況に応じて手法を選ぶ必要がある
- ・人の労力削減や機械学習モデルの性能向上など複数の観点から, ハイパパラメータ最適化をはじめとする, 機械学習の自動化を支える技術は向上とともに拡大していくと考えられる.

はじめに

ハイパパラメータ
最適化

グリッドサーチ

ランダムサーチ

ベイズ最適化

Nelder-Mead 法

グレーボックス最
適化

最適化手法選択

まとめ