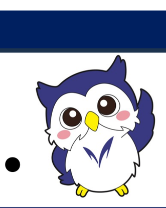
確率的勾配降下法の平滑化効果を利用した段階的最適化アルゴリズムによる経験損失最小化問題のための大域的最適化

佐藤 尚樹, 飯塚 秀明 (明治大学)

要約：暗黙的な段階的最適化で経験損失関数(非凸関数)の大域的最適化を実現できる。



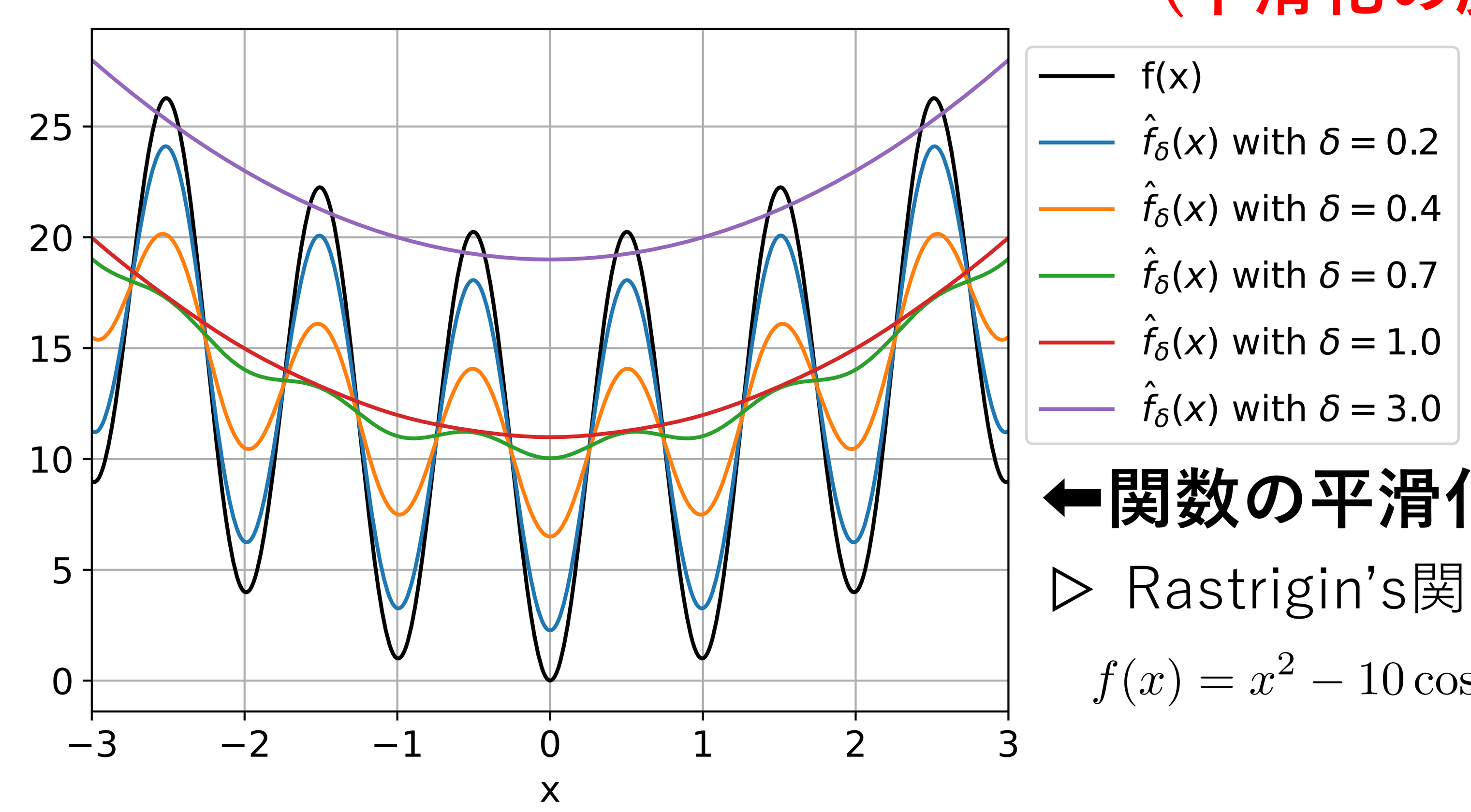
Ⅰ. 段階的最適化（Graduated Optimization）

▷徐々に小さくなるノイズで平滑化された関数の列を順番に最適化する大域的最適化手法.

▷関数 $f\colon \mathbb{R}^d \to \mathbb{R}$ を, 大きさ $\delta \in \mathbb{R}$ のノイズで平滑化した関数は，次のように定義される.

$$\hat{f}_\delta(\boldsymbol{x}) := \mathbb{E}_{\boldsymbol{u}\sim\mathcal{N}\left(\boldsymbol{0},\frac{1}{\sqrt{d}}I_d\right)}\left[f(\boldsymbol{x}-\underline{\delta}\boldsymbol{u})\right]$$

ノイズスケール（平滑化の度合い）



⬅関数の平滑化の例

▷ Rastrigin's関数(1変数)

$$f(x) = x^2 - 10\cos(2\pi x) + 10$$

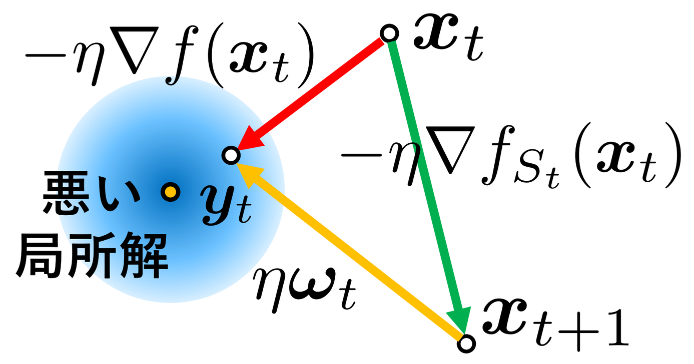
▷σ-nice関数[Hazan et al., 2016]

(i)任意の $\delta_m > 0$ と $\boldsymbol{x}^\star_{\delta_m}$ に対して, $\|\boldsymbol{x}^\star_{\delta_m} - \boldsymbol{x}^\star_{\delta_{m+1}}\| \leq \delta_{m+1} := \frac{\delta_m}{2}$.

(ii)任意の $\delta_m > 0$ に対して, 関数 $\hat{f}_{\delta_m}$ は近傍 $N(\boldsymbol{x}^\star_{\delta_m}; 3\delta_m)$ でσ-強凸である.

Ⅱ.確率的勾配降下法の平滑化特性

▷時刻 t のパラメータを $\boldsymbol{x}_t$ とする．最急降下法で更新した先を $\boldsymbol{y}_t$, SGDで更新した先を $\boldsymbol{x}_{t+1}$ とすると,



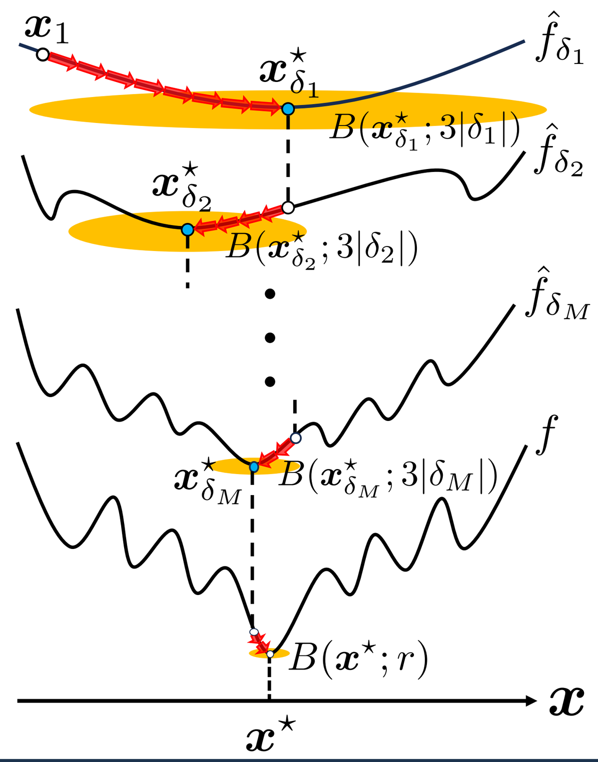
(GD) $\boldsymbol{y}_t := \boldsymbol{x}_t - \eta\nabla f(\boldsymbol{x}_t)$

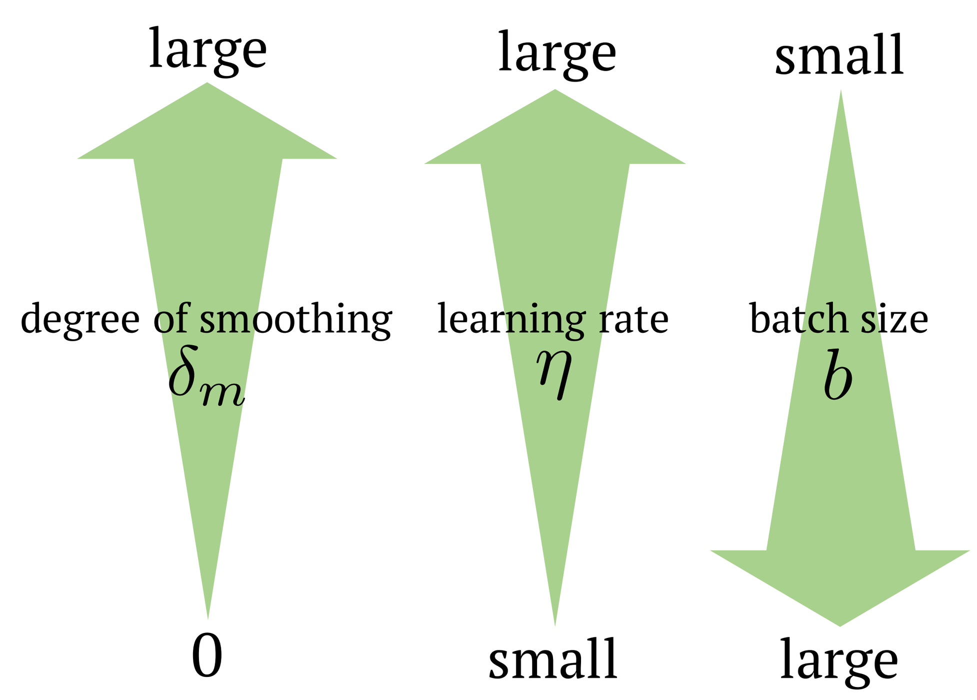
(SGD) $\boldsymbol{x}_{t+1} := \boldsymbol{x}_t - \eta\nabla f_{S_t}(\boldsymbol{x}_t)$

確率的ノイズ $\boldsymbol{\omega}_t := \underbrace{\nabla f_{S_t}(\boldsymbol{x}_t)}_{\text{SGDの探索方向}} - \underbrace{\nabla f(\boldsymbol{x}_t)}_{\text{GDの探索方向（最急降下方向）}}$

$$\begin{aligned}\mathbb{E}_{\boldsymbol{\omega}_t}[\boldsymbol{y}_{t+1}] &= \mathbb{E}_{\boldsymbol{\omega}_t}[\boldsymbol{y}_t] - \eta\nabla\mathbb{E}_{\boldsymbol{\omega}_t}[f(\boldsymbol{y}_t - \eta\boldsymbol{\omega}_t)] \\ &= \mathbb{E}_{\boldsymbol{\omega}_t}[\boldsymbol{y}_t] - \eta\nabla\mathbb{E}_{\boldsymbol{u}\sim\mathcal{N}\left(\boldsymbol{0};\frac{1}{\sqrt{d}}I_d\right)}\left[f\left(\boldsymbol{y}_t - \frac{\eta C}{\sqrt{b}}\boldsymbol{u}_t\right)\right] \\ &= \mathbb{E}_{\boldsymbol{\omega}_t}[\boldsymbol{y}_t] - \eta\nabla\hat{f}_{\frac{\eta C}{\sqrt{b}}}(\boldsymbol{y}_t)\end{aligned}$$

が成り立つから，関数 f をSGDで最適化することと，関数 $\hat{f}_{\frac{\eta C}{\sqrt{b}}}$ をGDで最適化することは期待値の意味で等価.





Ⅲ. 暗黙的な段階的最適化（Implicit Graduated Optimization）

Algorithm 1 Implicit Graduated Optimization

Require: $\epsilon, \boldsymbol{x}_1 \in B(\boldsymbol{x}^\star_{\delta_1}; 3\delta_1), \eta_1 > 0, b_1 \in [n], \gamma \geq 0.5$
  $\delta_1 := \frac{\eta_1 C}{\sqrt{b_1}}, \alpha_0 := \min\left\{\frac{1}{16L_f\delta_1}, \frac{1}{\sqrt{2\sigma\delta_1}}\right\}, M := \log_\gamma \alpha_0\epsilon$
  for $m = 1$ to $M+1$ **do**
    if $m \neq M+1$ **then**
      $\epsilon_m := \sigma\delta_m^2/2,\ T_m := H_m/\epsilon_m$
      $\kappa_m/\sqrt{\lambda_m} = \gamma\ (\kappa_m \in (0,1], \lambda_m \geq 1)$
    end if
    $\boldsymbol{x}_{m+1} := \mathrm{GD}(T_m, \boldsymbol{x}_m, \hat{f}_{\delta_m}, \eta_m)$
    $\eta_{m+1} := \kappa_m\eta_m, b_{m+1} := \lambda_m b_m$
    $\delta_{m+1} := \frac{\eta_{m+1}C}{\sqrt{b_{m+1}}}$
  end for
  return $\boldsymbol{x}_{M+2}$

M個の平滑化された関数の最適化に使われる.

Algorithm 2 Gradient Descent

Require: $T_m, \hat{\boldsymbol{x}}^{(m)}_1, \hat{f}_{\delta_m}, \eta > 0$
  for $t = 1$ to T_m **do**
    $\hat{\boldsymbol{x}}^{(m)}_{t+1} := \hat{\boldsymbol{x}}^{(m)}_t - \eta\nabla\hat{f}_{\delta_m}(\boldsymbol{x}_t)$
  end for
  return $\hat{\boldsymbol{x}}^{(m)}_{T_m+1}$

▷ $\mathcal{O}\left(1/\epsilon^2\right)$回の反復で任意の$\sigma$-nice関数 f の大域的最適解のϵ-近傍に収束する.

▷ SGDよりも高いテスト精度と低い損失を達成.

Ⅳ. 平滑化の度合い, Sharpness, 汎化性能の関係

▷ CIFAR100データセットによるResNet18の訓練でそれぞれを計測した.

▷ 学習率 $\eta \in${0.01, 0.05, 0.1, 0.5}, バッチサイズb∈{2, 4, ⋯, 8192}, 200epochs