Explicit and Implicit Graduated Optimization in Deep Neural Networks

Naoki Sato, Hideaki Iiduka
Meiji University

Conclusion: SGD's stochastic noise allows global optimization of nonconvex functions.

I. Explicit Graduated Optimization

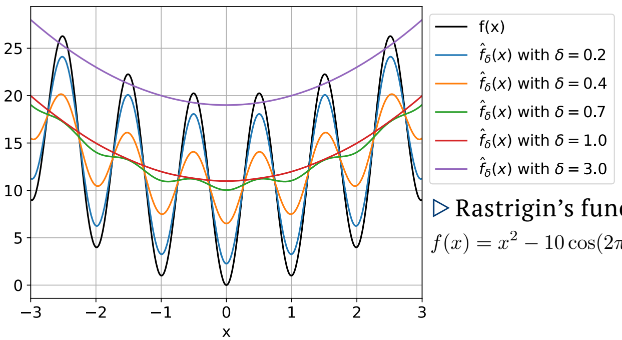
▷ Graduated Optimization is a global optimization technique that optimizes a sequence of functions smoothed by progressively smaller noise in order.

▷ Smoothing of a function is achieved by convolving the function with a random variable that follows a normal distribution or a uniform distribution.

Definition 1 (Smoothed function).
Given a function f, define $\hat{f}_\delta$ to be the function obtained by smoothing f as

$$\hat{f}_\delta(\boldsymbol{x}) := \mathbb{E}_{\boldsymbol{u}\sim B(\boldsymbol{0};1)}\left[f(\boldsymbol{x}-\delta\boldsymbol{u})\right],$$

where $\delta \in \mathbb{R}$ represents the degree of smoothing and $\boldsymbol{u}$ is a random variable distributed uniformly over Euclidean closed ball $B(\boldsymbol{0};1)$.



▷ Rastrigin's function
$f(x) = x^2 - 10\cos(2\pi x) + 10$

Algorithm 1 Explicit Graduated Optimization

Require: $\delta_1 > 0, M \in \mathbb{N}, \boldsymbol{x}_1 \in \mathbb{R}^d$
  for $m = 1$ to $M+1$ **do**
    $\hat{f}_{\delta_m}(\boldsymbol{x}) := \mathbb{E}_{\boldsymbol{u}\sim B(\boldsymbol{0};1)}\left[f(\boldsymbol{x}-\delta\boldsymbol{u})\right]$
    $\boldsymbol{x}_{m+1} := \mathrm{GD}(\boldsymbol{x}_m, \hat{f}_{\delta_m})$
    $\delta_{m+1} := \delta_m/2$
  end for
  return $\boldsymbol{x}_{M+2}$

Algorithm 2 Gradient Descent

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

Step 1. prepare smoothed function.
Step 2. optimize smoothed function.
Step 3. update the degree of smoothing.

▷ Since computing the integral of the empirical loss function is not easy, applying this algorithm to the training of DNNs is not practical.

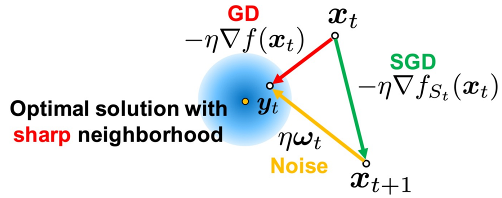
II. SGD's smoothing property

▷ Unlike gradient descent (GD), stochastic gradient descent (SGD) processes only b data points simultaneously, so there is stochastic noise at each time:

$$\boldsymbol{\omega}_t := \nabla f_{S_t}(\boldsymbol{x}_t) - \nabla f(\boldsymbol{x}_t)$$

▷ At time t, let $\boldsymbol{y}_t$ be the parameter updated by GD and $\boldsymbol{x}_{t+1}$ be the parameter updated by SGD, i.e.,

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



▷ Then, the following holds:

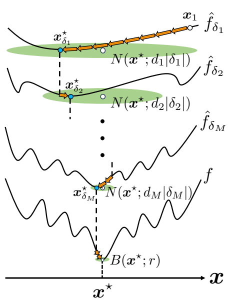
$$\mathbb{E}_{\boldsymbol{\omega}_t}[\boldsymbol{y}_{t+1}] = \mathbb{E}_{\boldsymbol{\omega}_t}[\boldsymbol{y}_t] - \eta\nabla\hat{f}_{\frac{\eta C}{\sqrt{b}}}(\boldsymbol{y}_t),$$

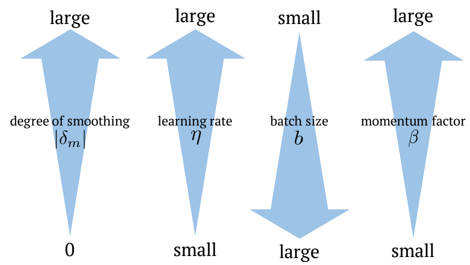
where C^2 is the variance of stochastic gradient.

▷ Therefore, optimizing the objective function f by SGD is equivalent to optimizing its smoothed version $\hat{f}_{\frac{\eta C}{\sqrt{b}}}$ by GD in the sense of expectation.

▷ From the same discussion for stochastic noise of SGD with momentum (NSHB), the degrees of smoothing by stochastic noise of SGD and NSHB are as follows:

$$\delta^{\mathrm{SGD}} = \eta\sqrt{\frac{C^2}{b}},\ \ \delta^{\mathrm{NSHB}} = \eta\sqrt{\frac{1}{1-\beta}\frac{C^2}{b}}.$$





III. Implicit Graduated Optimization

Definition 2 (new σ-nice function).
A function $f : \mathbb{R}^d \to \mathbb{R}$ is said to be "new σ-nice" if the following conditions hold:

(i) For all $m \in [M]$, there exist $\delta_m \in \mathbb{R}$ with $|\delta_{m+1}| := \gamma_m|\delta_m|$ and $\boldsymbol{x}^\star_{\delta_m}$ such that $\left\|\boldsymbol{x}^\star_{\delta_m} - \boldsymbol{x}^\star_{\delta_{m+1}}\right\| \leq |\delta_m| - |\delta_{m+1}|$.

(ii) For all $m \in [M]$ and all $\gamma_m \in (0,1)$, there exist $\delta_m \in \mathbb{R}$ with $|\delta_{m+1}| := \gamma_m|\delta_m|$ and $d_m > 1$ such that the function $\hat{f}_{\delta_m}(\boldsymbol{x})$ is σ-strongly convex on $N(\boldsymbol{x}^\star; d_m|\delta_m|)$.

Algorithm 3 Implicit Graduated Optimization with SGD

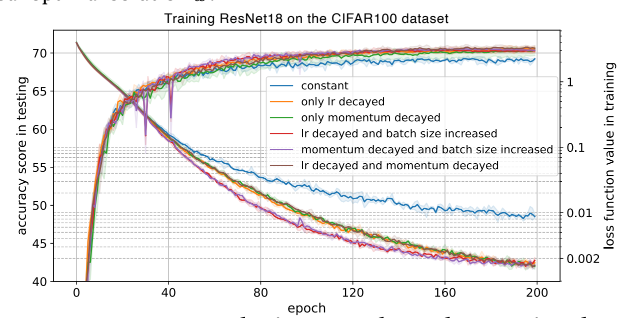
Require: $\epsilon, \boldsymbol{x}_1 \in B(\boldsymbol{x}^\star_{\delta_1}; 3\delta_1), \eta_1 > 0, b_1 \in [n], \gamma \geq 0.5$
  $\delta_1 := \frac{\eta_1 C}{\sqrt{b_1}}, \alpha_0 := \min\left\{\frac{1}{16L_f\delta_1}, \frac{1}{\sqrt{2\sigma}\delta_1}\right\}, M := \log_\gamma \alpha_0\epsilon$
 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

Decrease the degree of smoothing by decreasing the learning rate and increasing the batch size

Algorithm 2 Gradient Descent

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

Theorem 2 (Convergence Analysis of Algorithm 3).
Let f be a new σ-nice function. Suppose that we apply Algorithm 3; after $\mathcal{O}(1/\epsilon^2)$ rounds. Then, the algorithm reaches an ϵ-neighborhood of the global optimal solution $\boldsymbol{x}^\star$.



▷ Hence, common technique such as decreasing learning rate or momentum factor and increasing batch size actually contribute to the global optimization of the nonconvex objective function!