Adaptive Optimization Methods for Machine Learning and Signal Processing

Volkan Cevher — *volkan.cevher@epfl.ch*
Ali Kavis — *ali.kavis@epfl.ch*
Kfir Y. Levy — *kfirylevy@technion.ac.il*
Ahmet Alacaoglu — *ahmet.alacaoglu@epfl.ch*

Part II/IV: Introduction to adaptive first-order methods

Laboratory for Information and Inference Systems (LIONS)
École Polytechnique Fédérale de Lausanne (EPFL)

GD vs SGD

Consider the following optimization problem:

$$\boxed{\min_{x \in \mathcal{X}} f(x)} \tag{1}$$

GD vs SGD

Consider the following optimization problem:

$$\boxed{\min_{x \in \mathcal{X}} f(x)} \tag{1}$$

Update rule: (When $\mathcal{X} = \mathbb{R}^d$)

For $t = 1, \cdots, T$

$$x_{t+1} = x_t - \eta_t g_t \tag{2}$$

GD vs SGD

Consider the following optimization problem:

$$\boxed{\min_{x \in \mathcal{X}} f(x)} \tag{1}$$

Update rule: (When $\mathcal{X} = \mathbb{R}^d$)

For $t = 1, \cdots, T$

$$x_{t+1} = x_t - \eta_t g_t \tag{2}$$

Setting and gradient oracle:

- GD:

  $g_t = \nabla f(x_t)$ or $g_t \in \partial f(x_t)$

- SGD:

  $\mathbb{E}[g_t | x_t] = \nabla f(x_t)$ or $\mathbb{E}[g_t | x_t] \in \partial f(x_t)$

GD vs SGD

Consider the following optimization problem:

$$\boxed{\min_{x \in \mathcal{X}} f(x)} \tag{1}$$

Update rule: (When $\mathcal{X} = \mathbb{R}^d$)

For $t = 1, \cdots, T$

$$x_{t+1} = x_t - \eta_t g_t \tag{2}$$

Setting and gradient oracle:

- GD:

  $g_t = \nabla f(x_t)$ or $g_t \in \partial f(x_t)$

- SGD:

  $\mathbb{E}[g_t | x_t] = \nabla f(x_t)$ or $\mathbb{E}[g_t | x_t] \in \partial f(x_t)$

Notion of convergence:

- f is convex:

  $f(x_T) - f(x^*)$ or $\mathbb{E}[f(x_T) - f(x^*)]$

- f is non-convex:

  $\|\nabla f(x_T)\|^2$ or $\mathbb{E}[\|\nabla f(x_T)\|^2]$

Convergence in the convex setting

	$f(\cdot)$	oracle	step size	convergence rate
GD	L-smooth	$g_t = \nabla f(x_t)$	$\eta_t < \frac{1}{L}$	$\mathcal{O}\left(\frac{1}{T}\right)$[Nesterov, 2004]
GD	non-smooth	$g_t \in \partial f(x_t)$	$\eta_t = \mathcal{O}\left(\frac{1}{\sqrt{t}}\right)$	$\mathcal{O}\left(\frac{\log(T)}{\sqrt{T}}\right)$[Nesterov, 2004]
SGD	L-smooth	$\mathbb{E}[g_t \vert x_t] = \nabla f(x_t)$	$\eta_t = \mathcal{O}\left(\frac{1}{\sqrt{t}}\right)$	$\mathcal{O}\left(\frac{\log(T)}{\sqrt{T}}\right)$ [Lan, 2020]
SGD	non-smooth	$\mathbb{E}[g_t \vert x_t] \in \partial f(x_t)$	$\eta_t = \mathcal{O}\left(\frac{1}{\sqrt{t}}\right)$	$\mathcal{O}\left(\frac{\log(T)}{\sqrt{T}}\right)$ [Lan, 2020]

Convergence in the convex setting

	$f(\cdot)$	oracle	step size	convergence rate
GD	L-smooth	$g_t = \nabla f(x_t)$	$\eta_t < \frac{1}{L}$	$\mathcal{O}\left(\frac{1}{T}\right)$ [Nesterov, 2004]
GD	non-smooth	$g_t \in \partial f(x_t)$	$\eta_t = \mathcal{O}\left(\frac{1}{\sqrt{t}}\right)$	$\mathcal{O}\left(\frac{\log(T)}{\sqrt{T}}\right)$ [Nesterov, 2004]
SGD	L-smooth	$\mathbb{E}[g_t \vert x_t] = \nabla f(x_t)$	$\eta_t = \mathcal{O}\left(\frac{1}{\sqrt{t}}\right)$	$\mathcal{O}\left(\frac{\log(T)}{\sqrt{T}}\right)$ [Lan, 2020]
SGD	non-smooth	$\mathbb{E}[g_t \vert x_t] \in \partial f(x_t)$	$\eta_t = \mathcal{O}\left(\frac{1}{\sqrt{t}}\right)$	$\mathcal{O}\left(\frac{\log(T)}{\sqrt{T}}\right)$ [Lan, 2020]

Convergence in the convex setting

	$f(\cdot)$	oracle	step size	convergence rate
GD	L-smooth	$g_t = \nabla f(x_t)$	$\eta_t < \frac{1}{L}$	$\mathcal{O}\left(\frac{1}{T}\right)$ [Nesterov, 2004]
GD	non-smooth	$g_t \in \partial f(x_t)$	$\eta_t = \mathcal{O}\left(\frac{1}{\sqrt{T}}\right)$	$\mathcal{O}\left(\frac{\log(T)}{\sqrt{T}}\right)$ [Nesterov, 2004]
SGD	L-smooth	$\mathbb{E}[g_t \| x_t] = \nabla f(x_t)$	$\eta_t = \mathcal{O}\left(\frac{1}{\sqrt{T}}\right)$	$\mathcal{O}\left(\frac{\log(T)}{\sqrt{T}}\right)$ [Lan, 2020]
SGD	non-smooth	$\mathbb{E}[g_t \| x_t] \in \partial f(x_t)$	$\eta_t = \mathcal{O}\left(\frac{1}{\sqrt{T}}\right)$	$\mathcal{O}\left(\frac{\log(T)}{\sqrt{T}}\right)$ [Lan, 2020]

Convergence in the convex setting

	$f(\cdot)$	oracle	step size	convergence rate
GD	L-smooth	$g_t = \nabla f(x_t)$	$\eta_t < \frac{1}{L}$	$\mathcal{O}\left(\frac{1}{T}\right)$ [Nesterov, 2004]
GD	non-smooth	$g_t \in \partial f(x_t)$	$\eta_t = \mathcal{O}\left(\frac{1}{\sqrt{T}}\right)$	$\mathcal{O}\left(\frac{1}{\sqrt{T}}\right)$ [Nesterov, 2004]
SGD	L-smooth	$\mathbb{E}[g_t \vert x_t] = \nabla f(x_t)$	$\eta_t = \mathcal{O}\left(\frac{1}{\sqrt{T}}\right)$	$\mathcal{O}\left(\frac{1}{\sqrt{T}}\right)$ [Lan, 2020]
SGD	non-smooth	$\mathbb{E}[g_t \vert x_t] \in \partial f(x_t)$	$\eta_t = \mathcal{O}\left(\frac{1}{\sqrt{T}}\right)$	$\mathcal{O}\left(\frac{1}{\sqrt{T}}\right)$ [Lan, 2020]

Convergence in the convex setting

	$f(\cdot)$	oracle	step size	convergence rate
GD	L-smooth	$g_t = \nabla f(x_t)$	$\eta_t < \frac{1}{L}$	$\mathcal{O}\left(\frac{1}{T}\right)$ [Nesterov, 2004]
GD	non-smooth	$g_t \in \partial f(x_t)$	$\eta_t = \mathcal{O}\left(\frac{1}{\sqrt{t}}\right)$	$\mathcal{O}\left(\frac{\log(T)}{\sqrt{T}}\right)$ [Nesterov, 2004]
SGD	L-smooth	$\mathbb{E}[g_t \mid x_t] = \nabla f(x_t)$	$\eta_t = \mathcal{O}\left(\frac{1}{\sqrt{t}}\right)$	$\mathcal{O}\left(\frac{\log(T)}{\sqrt{T}}\right)$ [Lan, 2020]
SGD	non-smooth	$\mathbb{E}[g_t \mid x_t] \in \partial f(x_t)$	$\eta_t = \mathcal{O}\left(\frac{1}{\sqrt{t}}\right)$	$\mathcal{O}\left(\frac{\log(T)}{\sqrt{T}}\right)$ [Lan, 2020]

Observations

- GD uses the **worst-case, global** constant for selecting step size
- SGD uses a **fixed, pre-determined** step size routine

Convergence in the convex setting

	$f(\cdot)$	oracle	step size	convergence rate
GD	L-smooth	$g_t = \nabla f(x_t)$	$\eta_t < \frac{1}{L}$	$\mathcal{O}\left(\frac{1}{T}\right)$ [Nesterov, 2004]
GD	non-smooth	$g_t \in \partial f(x_t)$	$\eta_t = \mathcal{O}\left(\frac{1}{\sqrt{t}}\right)$	$\mathcal{O}\left(\frac{\log(T)}{\sqrt{T}}\right)$ [Nesterov, 2004]
SGD	L-smooth	$\mathbb{E}[g_t \mid x_t] = \nabla f(x_t)$	$\eta_t = \mathcal{O}\left(\frac{1}{\sqrt{t}}\right)$	$\mathcal{O}\left(\frac{\log(T)}{\sqrt{T}}\right)$ [Lan, 2020]
SGD	non-smooth	$\mathbb{E}[g_t \mid x_t] \in \partial f(x_t)$	$\eta_t = \mathcal{O}\left(\frac{1}{\sqrt{t}}\right)$	$\mathcal{O}\left(\frac{\log(T)}{\sqrt{T}}\right)$ [Lan, 2020]

Observations

- GD uses the **worst-case, global** constant for selecting step size
- SGD uses a **fixed, pre-determined** step size routine

How could we customize step size based on the **local** information?

Template for adaptive methods

Algorithm: Adaptive First-Order Methods Template

Input: Iterations T; $x_1 \in \mathcal{X} \subset \mathbb{R}^d$
for $t = 1, ..., T$ **do**
    Obtain a gradient estimate g_t

    $x_{t+1} = P_{\mathcal{X}}\left(x_t - \eta_t g_t\right)$
end for

→ Orthogonal projection onto $\mathcal{X}$: $P_{\mathcal{X}}(x) = \arg\min_{z \in \mathcal{X}} \|z - x\|^2$

Properties:

Template for adaptive methods

Algorithm: Adaptive First-Order Methods Template

Input: Iterations T; $x_1 \in \mathcal{X} \subset \mathbb{R}^d$
for $t = 1, ..., T$ **do**
  Obtain a gradient estimate g_t
  Compute $m_t = h_1(g_1, \cdots, g_t)$

  $x_{t+1} = P_{\mathcal{X}}\left(x_t - \eta_t m_t\right)$
end for

$\rightarrow$ Orthogonal projection onto $\mathcal{X}$: $P_{\mathcal{X}}(x) = \arg\min_{z \in \mathcal{X}} \|z - x\|^2$

Properties:

$m_t \in \mathbb{R}^d$: **first-order estimate:**

- computes a (negative) descent direction
- GD & SGD: $m_t = g_t$

Template for adaptive methods

Algorithm: Adaptive First-Order Methods Template

Input: Iterations T; $x_1 \in \mathcal{X} \subset \mathbb{R}^d$
for $t = 1, ..., T$ **do**
  Obtain a gradient estimate g_t
  Compute $m_t = h_1(g_1, \cdots, g_t)$
  Compute $H_t = h_2(g_1, \cdots, g_t)$
  $x_{t+1} = P_{\mathcal{X}}^{H_t}\left(x_t - \eta H_t^{-1} m_t\right)$
end for

$\rightarrow$ Metric projection onto $\mathcal{X}$: $P_{\mathcal{X}}^{H}(x) = \arg\min_{z \in \mathcal{X}} \langle z - x, H(z - x) \rangle$

Properties:

$m_t \in \mathbb{R}^d$: **first-order estimate:**

- computes a (negative) descent direction
- GD & SGD: $m_t = g_t$

$H_t \in \mathbb{R}^{d \times d}$: **second-order estimate:**

- accumulates outer products, i.e., $g_t g_t^\top$
- per coordinate step size
- GD & SGD: $H_t = I_d$

AdaGrad & AdaGrad-Scalar

Algorithm: AdaGrad [Duchi et al., 2011]

Input: Iterations T; $x_1 \in \mathcal{X} \subset \mathbb{R}^d$; $Q_0 = \mathbf{0}^{d \times d}$
for $t = 1, \ldots, T$ **do**
    Obtain a gradient estimate g_t
    $Q_t = Q_{t-1} + g_t g_t^\top$
    Compute $H_t = \sqrt{\mathrm{diag}(Q_t)}$
    $x_{t+1} = P_{\mathcal{X}}^{H_t}\left(x_t - \eta H_t^{-1} g_t\right)$
end for

Algorithm: AdaGrad-Scalar

Input: Iterations T; $x_1 \in \mathcal{X} \subset \mathbb{R}^d$; $s_t = 0$
for $t = 1, \ldots, T$ **do**
    Obtain a gradient estimate g_t
    $Q_t = Q_{t-1} + \|g_t\|^2$
    Compute $H_t = \sqrt{Q_t}$
    $x_{t+1} = P_{\mathcal{X}}\left(x_t - \eta H_t^{-1} g_t\right)$
end for

AdaGrad & AdaGrad-Scalar

Algorithm: AdaGrad [Duchi et al., 2011]

Input: Iterations T; $x_1 \in \mathcal{X} \subset \mathbb{R}^d$; $Q_0 = \mathbf{0}^{d \times d}$
for $t = 1, ..., T$ **do**
  Obtain a gradient estimate g_t
  $Q_t = Q_{t-1} + g_t g_t^\top$
  Compute $H_t = \sqrt{\mathrm{diag}(Q_t)}$
  $x_{t+1} = P_{\mathcal{X}}^{H_t}\left(x_t - \eta H_t^{-1} g_t\right)$
end for

Algorithm: AdaGrad-Scalar

Input: Iterations T; $x_1 \in \mathcal{X} \subset \mathbb{R}^d$; $s_t = 0$
for $t = 1, ..., T$ **do**
  Obtain a gradient estimate g_t
  $Q_t = Q_{t-1} + \|g_t\|^2$
  Compute $H_t = \sqrt{Q_t}$
  $x_{t+1} = P_{\mathcal{X}}\left(x_t - \eta H_t^{-1} g_t\right)$
end for

$$H_{t,i} = \sqrt{\sum_{\tau=1}^{t} g_{\tau,i}^2}$$

$$x_{t+1,i} = x_{t,i} - \frac{\eta}{\sqrt{\sum_{\tau=1}^{t} g_{\tau,i}^2}} g_{t,i}$$

$$H_t = \sqrt{\sum_{\tau=1}^{t} \|g_\tau\|^2}$$

$$x_{t+1} = x_t - \frac{\eta}{\sqrt{\sum_{\tau=1}^{t} \|g_\tau\|^2}} g_t$$

How to make sense of AdaGrad step size?

Optimization problem:

$$\boxed{\min_{x \in \mathcal{X}} f(x)}$$

How to make sense of AdaGrad step size?

Optimization problem:

$$\boxed{\min_{x \in \mathcal{X}} f(x)}$$

Assumptions:

- $f : \mathbb{R}^d \to \mathbb{R}$ is non-smooth and convex
- $\mathcal{X} \subset \mathbb{R}^d$ is compact and convex
- Define $D = \max_{x,y \in \mathcal{X}} \|x - y\|$ as diameter of $\mathcal{X}$.

How to make sense of AdaGrad step size?

Optimization problem:

$$\boxed{\min_{x \in \mathcal{X}} f(x)}$$

Assumptions:

- $f : \mathbb{R}^d \to \mathbb{R}$ is non-smooth and convex
- $\mathcal{X} \subset \mathbb{R}^d$ is compact and convex
- Define $D = \max_{x,y \in \mathcal{X}} \|x - y\|$ as diameter of $\mathcal{X}$.

AdaGrad-Scalar:

$$x_{t+1} = P_{\mathcal{X}}\left(x_t - \frac{D}{\sqrt{\sum_{\tau=1}^{t} \|\nabla f(x_t)\|^2}} \nabla f(x_t) \right)$$

How to make sense of AdaGrad step size?

Optimization problem:

$$\boxed{\min_{x \in \mathcal{X}} f(x)}$$

Assumptions:

- $f : \mathbb{R}^d \to \mathbb{R}$ is non-smooth and convex
- $\mathcal{X} \subset \mathbb{R}^d$ is compact and convex
- Define $D = \max_{x,y \in \mathcal{X}} \|x - y\|$ as diameter of $\mathcal{X}$.

AdaGrad-Scalar:

$$x_{t+1} = P_{\mathcal{X}}\left(x_t - \frac{D}{\sqrt{\sum_{\tau=1}^{t} \|\nabla f(x_t)\|^2}} \nabla f(x_t) \right)$$

High-level intuition:

- Large gradients observed $\implies$ step size decays faster
- Small gradients observed $\implies$ step size stabilizes

Abridged proof of AdaGrad step size

$$\boxed{x_{t+1} = P_{\mathcal{X}}\left(x_t - \eta_t \nabla f(x_t)\right)}$$

Abridged proof of AdaGrad step size

$$\boxed{x_{t+1} = P_{\mathcal{X}}\left(x_t - \eta_t \nabla f(x_t)\right)}$$

Goal: Show value convergence rate $f(\bar{x}_T) - f(x^*)$, where $\bar{x}_T = \left(\sum_{t=1}^{T} x_t\right)/T$.

Key assumption: Assume that η_t is non-increasing.

Abridged proof of AdaGrad step size

$$\boxed{x_{t+1} = P_{\mathcal{X}}\left(x_t - \eta_t \nabla f(x_t)\right)}$$

Goal: Show value convergence rate $f(\bar{x}_T) - f(x^*)$, where $\bar{x}_T = \left(\sum_{t=1}^{T} x_t\right)/T$.

Key assumption: Assume that η_t is non-increasing.

1. Eventually, we arrive at the following bound.

$$f(\bar{x}_T) - f(x^*) \leq \frac{1}{T}\left(\frac{1}{2}\sum_{t=1}^{T} \eta_t \left\|\nabla f(x_t)\right\|^2 + \frac{D^2}{2\eta_T}\right)$$

Abridged proof of AdaGrad step size

$$\boxed{x_{t+1} = P_{\mathcal{X}}\left(x_t - \eta_t \nabla f(x_t)\right)}$$

Goal: Show value convergence rate $f(\bar{x}_T) - f(x^*)$, where $\bar{x}_T = \left(\sum_{t=1}^{T} x_t\right)/T$.

Key assumption: Assume that η_t is non-increasing.

1. Eventually, we arrive at the following bound.

$$f(\bar{x}_T) - f(x^*) \leq \frac{1}{T}\left(\frac{1}{2}\sum_{t=1}^{T} \eta_t \left\|\nabla f(x_t)\right\|^2 + \frac{D^2}{2\eta_T}\right)$$

2. Assume that $\eta_t = \eta$ is a constant step size. Minimize the upper bound with respect to η:

Abridged proof of AdaGrad step size

$$\boxed{x_{t+1} = P_{\mathcal{X}}\left(x_t - \eta_t \nabla f(x_t)\right)}$$

Goal: Show value convergence rate $f(\bar{x}_T) - f(x^*)$, where $\bar{x}_T = \left(\sum_{t=1}^T x_t\right)/T$.

Key assumption: Assume that η_t is non-increasing.

1. Eventually, we arrive at the following bound.

$$f(\bar{x}_T) - f(x^*) \leq \frac{1}{T}\left(\frac{1}{2}\sum_{t=1}^{T} \eta_t \left\|\nabla f(x_t)\right\|^2 + \frac{D^2}{2\eta_T}\right)$$

2. Assume that $\eta_t = \eta$ is a constant step size. Minimize the upper bound with respect to η:

$$\eta = \frac{D}{\sqrt{\sum_{t=1}^T \left\|\nabla f(x_t)\right\|^2}}$$

Abridged proof of AdaGrad step size

$$\boxed{x_{t+1} = P_{\mathcal{X}}\left(x_t - \eta_t \nabla f(x_t)\right)}$$

Goal: Show value convergence rate $f(\bar{x}_T) - f(x^*)$, where $\bar{x}_T = \left(\sum_{t=1}^{T} x_t\right)/T$.

Key assumption: Assume that η_t is non-increasing.

1. Eventually, we arrive at the following bound.

$$f(\bar{x}_T) - f(x^*) \leq \frac{1}{T}\left(\frac{1}{2}\sum_{t=1}^{T} \eta_t \left\|\nabla f(x_t)\right\|^2 + \frac{D^2}{2\eta_T}\right)$$

2. Assume that $\eta_t = \eta$ is a constant step size. Minimize the upper bound with respect to η:

$$\eta = \frac{D}{\sqrt{\sum_{t=1}^{T} \left\|\nabla f(x_t)\right\|^2}}$$

$$f(\bar{x}_T) - f(x^*) \leq \frac{D\sqrt{\sum_{t=1}^{T} \left\|\nabla f(x_t)\right\|^2}}{T}$$

Proof of AdaGrad step size (continued)

$$\boxed{x_{t+1} = P_{\mathcal{X}}\left(x_t - \eta_t \nabla f(x_t)\right)}$$

3. Let's pick a more realistic choice, $\eta_t = \frac{D}{\sqrt{2\sum_{\tau=1}^{t} \|\nabla f(x_\tau)\|^2}}$. Then,

Proof of AdaGrad step size (continued)

$$\boxed{x_{t+1} = P_{\mathcal{X}}\left(x_t - \eta_t \nabla f(x_t)\right)}$$

3. Let's pick a more realistic choice, $\eta_t = \frac{D}{\sqrt{2\sum_{\tau=1}^{t} \|\nabla f(x_\tau)\|^2}}$. Then,

$$f(\bar{x}_T) - f(x^*) \leq \frac{D\sqrt{2\sum_{t=1}^{T} \|\nabla f(x_t)\|^2}}{T}$$

Proof of AdaGrad step size (continued)

$$\boxed{x_{t+1} = P_{\mathcal{X}}\left(x_t - \eta_t \nabla f(x_t)\right)}$$

3. Let's pick a more realistic choice, $\eta_t = \frac{D}{\sqrt{2\sum_{\tau=1}^{t} \|\nabla f(x_\tau)\|^2}}$. Then,

$$f(\bar{x}_T) - f(x^*) \leq \frac{D\sqrt{2\sum_{t=1}^{T} \|\nabla f(x_t)\|^2}}{T}$$

We computed the adaptive convergence bound for AdaGrad!

AdaGrad - Convergence in the convex setting

Theorem (AdaGrad - Deterministic, Convex, Nonsmooth) [Duchi et al., 2011]

Let f be a G-Lipschitz, convex function and let D be the diameter of $\mathcal{X}$. The sequence $\{x_t\}_{t=1}^T$ generated by AdaGrad ensures

$$f(\bar{x}_T) - f(x^*) \leq \frac{D\sqrt{2\sum_{t=1}^T \|\nabla f(x_t)\|^2}}{T} \leq \frac{DG\sqrt{2}}{\sqrt{T}}$$

AdaGrad - Convergence in the convex setting

Theorem (AdaGrad - Deterministic, Convex, Nonsmooth) [Duchi et al., 2011]

Let f be a G-Lipschitz, convex function and let D be the diameter of $\mathcal{X}$. The sequence $\{x_t\}_{t=1}^T$ generated by AdaGrad ensures

$$f(\bar{x}_T) - f(x^*) \leq \frac{D\sqrt{2\sum_{t=1}^T \|\nabla f(x_t)\|^2}}{T} \leq \frac{DG\sqrt{2}}{\sqrt{T}}$$

Theorem (AdaGrad-Scalar - Stochastic, Convex, Smooth) [Levy et al., 2018]

Let f be an L-smooth, convex function and let global minimizer x^* of f lie in $\mathcal{X}$. Under bounded variance assumption, $\mathbb{E}\left[\|g_t - \nabla f(x_t)\|^2 \,|x_t\right] \leq \sigma^2$, the sequence $\{x_t\}_{t=1}^T$ generated by AdaGrad-Scalar ensures

$$f(\bar{x}_T) - f(x^*) = \mathcal{O}\left(\frac{LD^2}{T} + \frac{\sigma D}{\sqrt{T}}\right)$$

AdaGrad - Convergence in the non-convex setting

$$x_{t+1} = x_t - \frac{\eta}{\sqrt{\sum_{\tau=1}^{t} \|g_\tau\|^2}} g_t$$

Theorem (AdaGrad-Scalar - Stochastic, Non-convex, Smooth) [Ward et al., 2019]

Let f be a L-smooth function with $f^* = \min_x f(x) > -\infty$. The sequence $\{x_t\}_{t=1}^{T}$ generated by AdaGrad-Scalar ensures that with probability $1 - \delta$

$$\min_{0 \leq t \leq T-1} \|\nabla f(x_t)\|^2 = \mathcal{O}\left(\frac{\sigma \left(\frac{f(x_0) - f^*}{\eta} \right) + \sigma^2 \log(T)}{\delta^{3/2} \sqrt{T}} \right)$$

AdaGrad - Convergence in the non-convex setting

$$x_{t+1} = x_t - \frac{\eta}{\sqrt{\sum_{\tau=1}^{t} \|g_\tau\|^2}} g_t$$

Theorem (AdaGrad-Scalar - Stochastic, Non-convex, Smooth) [Ward et al., 2019]

Let f be a L-smooth function with $f^* = \min_x f(x) > -\infty$. The sequence $\{x_t\}_{t=1}^T$ generated by AdaGrad-Scalar ensures that with probability $1 - \delta$

$$\min_{0 \le t \le T-1} \|\nabla f(x_t)\|^2 = \mathcal{O}\left(\frac{\sigma \left(\frac{f(x_0) - f^*}{\eta} \right) + \sigma^2 \log(T)}{\delta^{3/2} \sqrt{T}} \right)$$

AdaGrad - Convergence in the non-convex setting

$$x_{t+1} = x_t - \frac{\eta}{\sqrt{\sum_{\tau=1}^{t} \|g_\tau\|^2}} g_t$$

Theorem (AdaGrad-Scalar - Stochastic, Non-convex, Smooth) [Ward et al., 2019]

Let f be a L-smooth function with $f^* = \min_x f(x) > -\infty$. The sequence $\{x_t\}_{t=1}^T$ generated by AdaGrad-Scalar ensures that with probability $1 - \delta$

$$\min_{0 \leq t \leq T-1} \|\nabla f(x_t)\|^2 = \mathcal{O}\left(\frac{\sigma \left(\frac{f(x_0) - f^*}{\eta} \right) + \sigma^2 \log(T)}{\delta^{3/2} \sqrt{T}} \right)$$

RmsProp

Algorithm: AdaGrad [Duchi et al., 2011]

Input: Iterations T; $x_1 \in \mathcal{X}$; $Q_0 = \mathbf{0}^{d \times d}$
for $t = 1, ..., T$ **do**
  Obtain a gradient estimate g_t
  $Q_t = Q_{t-1} + g_t g_t^T$
  Compute $H_t = \sqrt{\mathrm{diag}(Q_t)}$
  $x_{t+1} = P_{\mathcal{X}}^{H_t}\left(x_t - \eta H_t^{-1} g_t\right)$
end for

Algorithm: RMSProp [Hinton, 2012]

Input: Iterations T; $x_1 \in \mathcal{X}$; $Q_0 = \mathbf{0}$; $\beta \in (0, 1]$
for $t = 1, ..., T$ **do**
  Obtain a gradient estimate g_t
  $Q_t = \beta Q_{t-1} + (1 - \beta) g_t g_t^T$
  Compute $H_t = \sqrt{\mathrm{diag}(Q_t)}$
  $x_{t+1} = P_{\mathcal{X}}^{H_t}\left(x_t - \eta H_t^{-1} g_t\right)$
end for

RmsProp

Algorithm: AdaGrad [Duchi et al., 2011]

Input: Iterations T; $x_1 \in \mathcal{X}$; $Q_0 = \mathbf{0}^{d \times d}$
for $t = 1, \ldots, T$ **do**
  Obtain a gradient estimate g_t
  $Q_t = Q_{t-1} + g_t g_t^T$
  Compute $H_t = \sqrt{\mathrm{diag}(Q_t)}$
  $x_{t+1} = P_{\mathcal{X}}^{H_t}\left(x_t - \eta H_t^{-1} g_t\right)$
end for

Algorithm: RMSProp [Hinton, 2012]

Input: Iterations T; $x_1 \in \mathcal{X}$; $Q_0 = \mathbf{0}$; $\beta \in (0, 1]$
for $t = 1, \ldots, T$ **do**
  Obtain a gradient estimate g_t
  $Q_t = \beta Q_{t-1} + (1 - \beta) g_t g_t^T$
  Compute $H_t = \sqrt{\mathrm{diag}(Q_t)}$
  $x_{t+1} = P_{\mathcal{X}}^{H_t}\left(x_t - \eta H_t^{-1} g_t\right)$
end for

$$x_{t+1,i} = x_{t,i} - \frac{\eta}{\sqrt{\sum_{\tau=1}^{t} g_{\tau,i}^2}} g_{t,i}$$

$$x_{t+1,i} = x_{t,i} - \frac{\eta}{\sqrt{\sum_{\tau=1}^{t} \beta^{\tau-1}(1-\beta) g_{\tau,i}^2}} g_{t,i}$$

RmsProp

Algorithm: AdaGrad [Duchi et al., 2011]

1: **Input:** Iterations T; $x_1 \in \mathcal{X}$; $Q_0 = \mathbf{0}^{d\times d}$
2: **for** $t = 1, ..., T$ **do**
3: Obtain a gradient estimate g_t
4: $Q_t = Q_{t-1} + g_t g_t^T$
5: Compute $H_t = \sqrt{\mathrm{diag}(Q_t)}$
6: $x_{t+1} = P_{\mathcal{X}}^{H_t}\left(x_t - \eta H_t^{-1} g_t\right)$
7: **end for**

Algorithm: RMSProp [Hinton, 2012]

1: **Input:** Iterations T; $x_1 \in \mathcal{X}$; $Q_0 = \mathbf{0}$; $\beta \in (0, 1]$
2: **for** $t = 1, ..., T$ **do**
3: Obtain a gradient estimate g_t
4: $Q_t = \beta Q_{t-1} + (1-\beta) g_t g_t^T$
5: Compute $H_t = \sqrt{\mathrm{diag}(Q_t)}$
6: $x_{t+1} = P_{\mathcal{X}}^{H_t}\left(x_t - \eta H_t^{-1} g_t\right)$
7: **end for**

$$x_{t+1,i} = x_{t,i} - \frac{\eta}{\sqrt{\sum_{\tau=1}^{t} g_{\tau,i}^2}} g_{t,i}$$

$$x_{t+1,i} = x_{t,i} - \frac{\eta}{\sqrt{\sum_{\tau=1}^{t} \beta^{\tau-1}(1-\beta) g_{\tau,i}^2}} g_{t,i}$$

High-level intuition:

- Recent gradients receive larger weights.
- Consider a steep function, flat around minimum $\rightarrow$ better progress around minimum

Example: AdaGrad vs. RMSProp

Setting:

- $f(x) = x^4$ (one-dimensional function)
- $x_0 = 10$, $x^* = 0$

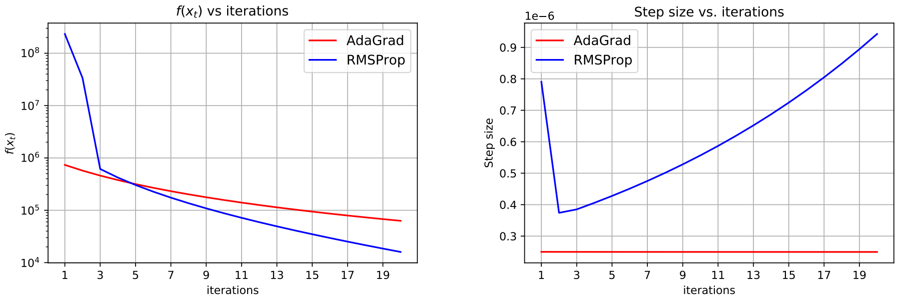

Figure: RMSProp vs. AdaGrad

Adam

Formula for Adam: RMSProp + first-order estimation $\implies$ Adam

Algorithm: RMSProp [Hinton, 2012]

1: **Input:** Iterations T; $x_1 \in \mathcal{X}$; $Q_0 = \mathbf{0}$; $\beta \in [0, 1]$
2: **for** $t = 1, ..., T$ **do**
3: Obtain a gradient estimate g_t
4: $Q_t = \beta Q_{t-1} + (1 - \beta) g_t g_t^T$
5: Compute $H_t = \sqrt{\mathrm{diag}(Q_t)}$
6: $x_{t+1} = P_{\mathcal{X}}^{H_t} \left(x_t - \eta H_t^{-1} g_t \right)$
7: **end for**

Algorithm: Adam [Kingma and Ba, 2014]

1: **Input:** Iterations T; $x_1 \in \mathcal{X}$; $Q_0 = \mathbf{0}$; $\beta_1, \beta_2 \in [0, 1)$
2: **for** $t = 1, ..., T$ **do**
3: Obtain a gradient estimate g_t
4: $m_t = \beta_1 m_{t-1} + (1 - \beta_1) g_t$
5: $Q_t = \beta_2 Q_{t-1} + (1 - \beta_2) g_t g_t^T$
6: Compute $H_t = \sqrt{\mathrm{diag}(Q_t)} + \epsilon$
7: $x_{t+1} = P_{\mathcal{X}}^{H_t} \left(x_t - \eta H_t^{-1} m_t \right)$
8: **end for**

[0] We ignore bias correction for Adam for simplicity

AmsGrad

Adam may converge as we expect!
When $\beta_1 < \sqrt{\beta_2}$, there exists a stochastic optimization problem for which Adam does not converge. [Reddi et al., 2018]

AmsGrad

Adam may converge as we expect!

When $\beta_1 < \sqrt{\beta_2}$, there exists a stochastic optimization problem for which Adam does not converge. [Reddi et al., 2018]

Algorithm: Adam [Kingma and Ba, 2014]

Input: Iterations T; $x_1 \in \mathcal{X}$; $Q_0 = \mathbf{0}$; $\beta_1, \beta_2 \in [0, 1)$
for $t = 1, ..., T$ **do**
  Obtain a gradient estimate g_t
  $m_t = \beta_1 m_{t-1} + (1 - \beta_1) g_t$
  $Q_t = \beta_2 Q_{t-1} + (1 - \beta_2) g_t g_t^T$
  Compute $H_t = \sqrt{\mathrm{diag}(Q_t)} + \epsilon$
  $x_{t+1} = P_{\mathcal{X}}^{H_t}\left(x_t - \eta_t H_t^{-1} m_t\right)$
end for

Algorithm: AmsGrad [Reddi et al., 2018]

Input: Iterations T; $x_1 \in \mathcal{X}$; $\{\beta_{1t}\}_{t=1}^{T}$; $\beta_2 \in [0, 1]$
Set: $Q_0, \hat{Q}_0 = \mathbf{0}$
for $t = 1, ..., T$ **do**
  Obtain a gradient estimate g_t
  $m_t = \beta_{1t} m_{t-1} + (1 - \beta_{1t}) g_t$
  $Q_t = \beta_2 Q_{t-1} + (1 - \beta_2) g_t g_t^T$
  $\hat{Q}_t = \max\left\{Q_t, \hat{Q}_{t-1}\right\}$
  Compute $H_t = \sqrt{\mathrm{diag}(\hat{Q}_t)} + \epsilon$
  $x_{t+1} = P_{\mathcal{X}}^{H_t}\left(x_t - \eta_t H_t^{-1} m_t\right)$
end for

AmsGrad: Convergence in the convex setting

Assumptions:

- Gradients are bounded across each coordinate, $G_\infty = \max_{x \in \mathcal{X}} \|\nabla f(x)\|_\infty$
- $\mathcal{X}$ has bounded diameter, $D_\infty = \max_{x,y \in \mathcal{X}} \|x - y\|_\infty$

AmsGrad: Convergence in the convex setting

Assumptions:

- Gradients are bounded across each coordinate, $G_\infty = \max_{x \in \mathcal{X}} \|\nabla f(x)\|_\infty$
- $\mathcal{X}$ has bounded diameter, $D_\infty = \max_{x,y \in \mathcal{X}} \|x - y\|_\infty$

Theorem (AmsGrad - Deterministic/Stochastic, Convex, Nonsmooth) [Reddi et al., 2018]

Let f be a G-Lipschitz, convex function, optimized over $\mathcal{X}$. Then, with $\eta_t = \eta / \sqrt{t}$, AmsGrad ensures,

$$f(\bar{x}_T) - f(x^*) = \mathcal{O}\left(\frac{\sqrt{\log(T)}}{\sqrt{T}} \right)$$

AmsGrad: Convergence in the non-convex setting

Theorem (AmsGrad - Stochastic, Non-convex, Smooth) [Alacaoglu et al., 2020]

Let f be L-smooth and non-convex. Assume that $\|g_t\|_\infty G_\infty$. Then, with $\eta_t = \eta / \sqrt{t}$ AmsGrad ensures,

$$\min_{t \in [T]} \mathbb{E}[\|\nabla f(x_t)\|^2] \leq \frac{1}{T} \sum_{t=1}^{T} \mathbb{E}[\|\nabla f(x_t)\|^2] = \mathcal{O}\left(\frac{\log(T)}{\sqrt{T}} \right)$$

AmsGrad: Convergence in the non-convex setting

Theorem (AmsGrad - Stochastic, Non-convex, Smooth) [Alacaoglu et al., 2020]

Let f be L-smooth and non-convex. Assume that $\|g_t\|_\infty G_\infty$. Then, with $\eta_t = \eta/\sqrt{t}$ AmsGrad ensures,

$$\min_{t\in[T]} \mathbb{E}[\|\nabla f(x_t)\|^2] \leq \frac{1}{T}\sum_{t=1}^{T}\mathbb{E}[\|\nabla f(x_t)\|^2] = \mathcal{O}\left(\frac{\log(T)}{\sqrt{T}}\right)$$

Theorem (AmsGrad - Stochastic, Non-convex, Smooth) [Zhou et al., 2018]

Let f be L-smooth and non-convex. Assume that $\|g_t\|_\infty G_\infty$. Then, with $\eta_t = \mathcal{O}\left(1/\sqrt{T}\right)$ AmsGrad ensures,

$$\min_{t\in[T]} \mathbb{E}[\|\nabla f(x_t)\|^2] \leq \frac{1}{T-1}\sum_{t=2}^{T}\mathbb{E}[\|\nabla f(x_t)\|^2] = \mathcal{O}\left(\frac{1}{\sqrt{T}}\right)$$

Example: Least squares with synthetic data

Setting:

- $f(x) = \|Ax - b\|^2$
- $A \in \mathbb{R}^{n \times d}$, $A \sim N(\mu, \sigma^2 I)$
- $n = 1000$, $d = 1000$

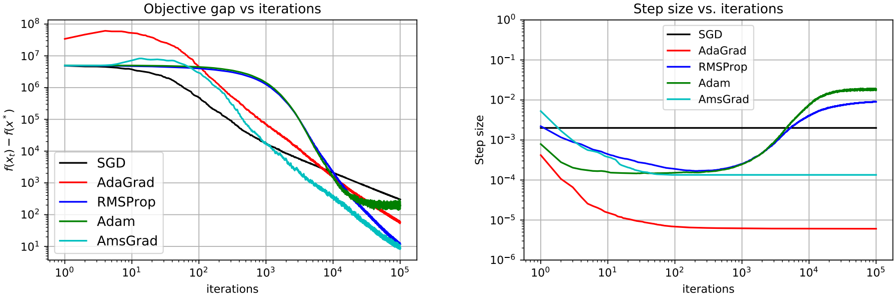

Figure: Comparison of convergence rate and stepsize evolution. Mini-batch stochastic gradients with a batch size of 20

Wrap up

- GD/SGD and their convergence under convexity
- General template for adaptive methods
- AdaGrad:
  - Intuition and small proof
  - Convergence for convex/non-convex problems
- RMSProp:
  - Intuition for second-order estimate computation
  - Comparison to AdaGrad
- Adam:
  - Intuition with respect to RMSProp
- AmsGrad:
  - Intuition with respect to Adam
  - Convergence for convex/non-convex problems

References I

[0] Alacaoglu, A., Malitsky, Y., Mertikopoulos, P., and Cevher, V. (2020).
A new regret analysis for Adam-type algorithms.
In III, H. D. and Singh, A., editors, *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pages 202–210. PMLR.

[0] Chang, C.-C. and Lin, C.-J. (2011).
LIBSVM: A library for support vector machines.
ACM Transactions on Intelligent Systems and Technology, 2:27:1–27:27.

[0] Duchi, J., Hazan, E., and Singer, Y. (2011).
Adaptive subgradient methods for online learning and stochastic optimization.
Journal of Machine Learning Research, 12(Jul):2121–2159.

[0] Hinton, G. (2012).
Lecture 6e rmsprop: Divide the gradient by a running average of its recent magnitude.

[0] Kingma, D. and Ba, J. (2014).
Adam: A method for stochastic optimization.
arXiv preprint arXiv:1412.6980.

References II

[0] Lan, G. (2020).
First-order and Stochastic Optimization Methods for Machine Learning.
Springer Series in the Data Sciences. Springer International Publishing.

[0] Levy, K., Yurtsever, A., and Cevher, V. (2018).
Online adaptive methods, universality and acceleration.
In *Proceedings of the 32nd International Conference on Neural Information Processing Systems*.

[0] Nesterov, Y. (2004).
Introductory Lectures on Convex Optimization: A Basic Course.
Kluwer, Boston, MA.

[0] Reddi, S. J., Kale, S., and Kumar, S. (2018).
On the convergence of adam and beyond.
In *International Conference on Learning Representations*.

[0] Ward, R., Wu, X., and Bottou, L. (2019).
AdaGrad stepsizes: Sharp convergence over nonconvex landscapes.
In Chaudhuri, K. and Salakhutdinov, R., editors, *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pages 6677–6686, Long Beach, California, USA. PMLR.

References III

[0] Zhou, D., Tang, Y., Yang, Z., Cao, Y., and Gu, Q. (2018).
On the convergence of adaptive gradient methods for nonconvex optimization.
CoRR, abs/1808.05671.