

Introduction to Machine Learning

Lec 03: Gradient Descent

Yaoliang Yu



UNIVERSITY OF
WATERLOO

FACULTY OF MATHEMATICS
**DAVID R. CHERITON SCHOOL
OF COMPUTER SCIENCE**

Sep 21, 2026

Problem

Unconstrained smooth minimization:

$$f_{\star} = \inf_{\mathbf{w} \in \mathbb{R}^d} f(\mathbf{w}).$$

- No constraint on the domain
- $f : \mathbb{R}^d \rightarrow \mathbb{R}$ is smooth, e.g. continuously differentiable
- f can be convex or nonconvex
- Minimizer may or may not be attained
- Maximization is just negation

Logistic Regression

$$\inf_{\mathbf{w}} \frac{1}{n} \sum_{i=1}^n \log[1 + \exp(-y_i \langle \mathbf{x}_i, \mathbf{w} \rangle)] \equiv \inf_{\mathbf{w}} \underbrace{\langle \log[1 + \exp(-\mathbf{w}\mathbf{A})], \frac{1}{n} \cdot \mathbf{1} \rangle}_{f(\mathbf{w})}$$

- $\mathbf{A} = [y_1 \mathbf{x}_1, \dots, y_n \mathbf{x}_n] \in \mathbb{R}^{p \times n}$
- $\mathbf{y} = [y_1, \dots, y_n] \in \{\pm 1\}^n$
- $\mathbf{w} \in \mathbb{R}^p$
- Again, f is (continuously) differentiable
- No constraint on $\mathbf{w}$

Example: Logistic Loss

$$f(\mathbf{w}) = \frac{1}{n} \sum_i \log[1 + \exp(-\langle \mathbf{a}_i, \mathbf{w} \rangle)]$$

$$\frac{df(\mathbf{w} + t\mathbf{z})}{dt} \Big|_{t=0} = \frac{1}{n} \sum_i \frac{\exp(-\langle \mathbf{a}_i, \mathbf{w} \rangle)}{1 + \exp(-\langle \mathbf{a}_i, \mathbf{w} \rangle)} \cdot (-\langle \mathbf{a}_i, \mathbf{z} \rangle)$$

$$\nabla f(\mathbf{w}) = \frac{1}{n} \cdot \frac{-\exp(-\mathbf{w}\mathbf{A})}{1 + \exp(-\mathbf{w}\mathbf{A})} \mathbf{A}^\top$$

- Recall $\mathbf{w} \in \mathbb{R}^p$, $\mathbf{A} \in \mathbb{R}^{p \times n}$
- What is the dimension of our gradient $\nabla f(\mathbf{w})$?

Gradient Descent

Algorithm 1: Richardson's first-order extrapolation for linear systems

Input: $\mathbf{w}_0 \in \mathbb{R}^d$, $A \in \mathbb{R}^{d \times d}$, $\mathbf{b} \in \mathbb{R}^d$

```
1 for  $t = 0, 1, \dots$  do
2    $\mathbf{g}_t \leftarrow A\mathbf{w}_t - \mathbf{b}$                                      // “gradient”
3    $\mathbf{w}_{t+1} \leftarrow \mathbf{w}_t - \eta_t \mathbf{g}_t$            //  $\eta_t$  is the step size
```

Algorithm 2: Gradient descent for unconstrained smooth minimization

Input: $\mathbf{w}_0 \in \mathbb{R}^d$, smooth function $f : \mathbb{R}^d \rightarrow \mathbb{R}$

```
1 for  $t = 0, 1, \dots$  do
2    $\mathbf{g}_t \leftarrow \nabla f(\mathbf{w}_t)$                              // compute the gradient
3    $\mathbf{w}_{t+1} \leftarrow \mathbf{w}_t - \eta_t \mathbf{g}_t$            //  $\eta_t$  is the step size
```

- Repeatedly subtract a multiple of the gradient

Intuition

$$\begin{aligned} f(\mathbf{w}_{t+1}) &= f(\mathbf{w}_t - \eta_t \nabla f(\mathbf{w}_t)) \\ &= f(\mathbf{w}_t) - \eta_t \langle \nabla f(\mathbf{w}_t), \nabla f(\mathbf{w}_t) \rangle + o(\eta_t) \\ &= f(\mathbf{w}_t) - \underbrace{\eta_t \|\nabla f(\mathbf{w}_t)\|_2^2}_{\geq 0} + o(\eta_t) \end{aligned}$$

- If $\nabla f(\mathbf{w}_t) = 0$, we are done
- Otherwise for small $\eta_t > 0$, we have $f(\mathbf{w}_{t+1}) < f(\mathbf{w}_t)$
- Strict improvement at each iteration; is it enough??

Lipschitz Continuity = Bounded Derivative

Theorem:

Let $T : \mathbb{R}^d \rightarrow \mathbb{R}^m$ be differentiable. Then, T is L -Lipschitz continuous:

$$\|T(\mathbf{w}) - T(\mathbf{z})\| \leq L\|\mathbf{w} - \mathbf{z}\|$$

if and only if

$$\sup_{\mathbf{w}} \|T'(\mathbf{w})\| = \sup_{\mathbf{w}} \sup_{\|\mathbf{z}\| \leq 1} \|T'(\mathbf{w})(\mathbf{z})\| \leq L.$$

- Lipschitz continuity: output change is bounded by input change
- Equivalently, derivative (i.e. infinitesimal change) is bounded

L-smoothness

We call a differentiable function $f : \mathbb{R}^d \rightarrow \mathbb{R}$ L-smooth if for all $\mathbf{w}$ and $\mathbf{z}$:

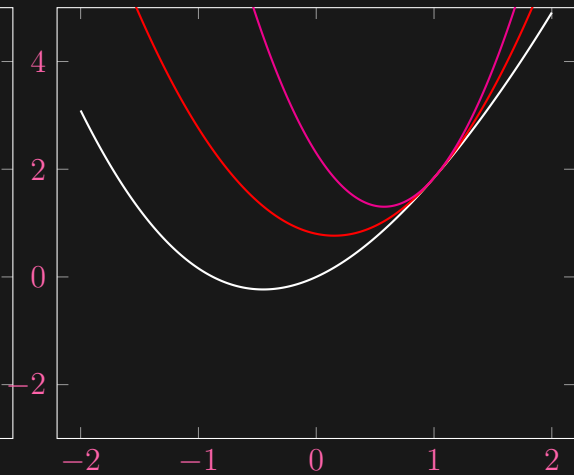
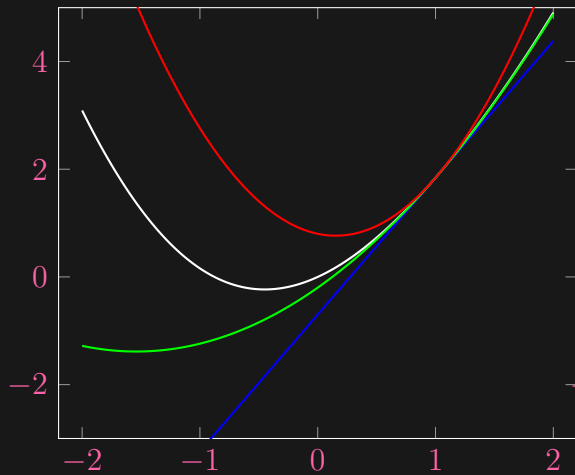
$$f(\mathbf{z}) \leq f(\mathbf{w}) + \underbrace{f'(\mathbf{w})(\mathbf{z} - \mathbf{w})}_{\langle \mathbf{z} - \mathbf{w}, \nabla f(\mathbf{w}) \rangle} + \frac{L}{2} \|\mathbf{z} - \mathbf{w}\|^2$$

Theorem: Characterizing L-smoothness

Consider the following statements for a real-valued smooth function:

- (I). **Vector-valued** derivative $f' : \mathbb{R}^d \rightarrow \mathbb{R}^d$ is L-Lipschitz continuous
- (II). **Matrix-valued** second-order derivative $f'' : \mathbb{R}^d \rightarrow \mathbb{R}^{d \times d}$ is L-bounded
- (III). **Real-valued** functions $\pm f$ are L-smooth

Then, (I) $\iff$ (II) $\implies$ (III). If f is convex or the norm is Euclidean, then all three are equivalent.



Importance of L -smoothness

$$f(\mathbf{w}) \leq f(\mathbf{w}_t) + \langle \mathbf{w} - \mathbf{w}_t, \nabla f(\mathbf{w}_t) \rangle + \frac{1}{2\eta_t} \|\mathbf{w} - \mathbf{w}_t\|_2^2$$

- RHS is a quadratic function of $\mathbf{w}$
- Inequality holds if $\eta_t \leq \frac{1}{L}$
- Minimize RHS w.r.t. $\mathbf{w}$:

$$\mathbf{w}_{t+1} \leftarrow \underset{\mathbf{w}}{\operatorname{argmin}} f(\mathbf{w}_t) + \frac{1}{2\eta_t} \|\mathbf{w} - [\mathbf{w}_t - \eta_t \nabla f(\mathbf{w}_t)]\|_2^2 - \frac{\eta_t}{2} \|\nabla f(\mathbf{w}_t)\|_2^2$$

- This is exactly gradient descent
- Moreover, $f(\mathbf{w}_{t+1}) \leq f(\mathbf{w}_t) - \frac{\eta_t}{2} \|\nabla f(\mathbf{w}_t)\|_2^2$

Example: Logistic Loss

$$f(\mathbf{w}) = \left\langle \log[1 + \exp(-\mathbf{w}\mathbf{A})], \frac{1}{n} \cdot \mathbf{1} \right\rangle$$

$$\nabla f(\mathbf{w}) = \frac{1}{n} \cdot \frac{-\exp(-\mathbf{w}\mathbf{A})}{1 + \exp(-\mathbf{w}\mathbf{A})} \mathbf{A}^\top = \frac{1}{n} \cdot \left[\frac{1}{1 + \exp(-\mathbf{w}\mathbf{A})} - 1 \right] \mathbf{A}^\top$$

$$\nabla^2 f(\mathbf{w}) = \frac{1}{n} \mathbf{A} \operatorname{diag} \left(\frac{\exp(-\mathbf{w}\mathbf{A})}{[1 + \exp(-\mathbf{w}\mathbf{A})]^2} \right) \mathbf{A}^\top \preceq \frac{1}{n} \mathbf{A} \mathbf{A}^\top$$

$$\sup_{\mathbf{w}} \|\nabla^2 f(\mathbf{w})\|_{\text{sp}} \leq \left\| \frac{1}{n} \mathbf{A} \mathbf{A}^\top \right\|_{\text{sp}} = \frac{1}{n} \|\mathbf{A}\|_{\text{sp}}^2 \leq \frac{1}{n} \|\mathbf{A}\|_{\text{F}}^2$$

Theorem: Convergence of gradient descent for L -smooth functions

Let $f : \mathbb{R}^d \rightarrow \mathbb{R}$ be L -smooth and bounded from below (i.e. $f_* > -\infty$). If the step size $\eta_t \in [\alpha, \frac{2}{L} - \beta]$ for some $\alpha, \beta > 0$, then the gradient descent iterate $\{\mathbf{w}_t\}$ satisfies $\nabla f(\mathbf{w}_t) \rightarrow \mathbf{0}$. Moreover,

$$\min_{0 \leq t \leq T-1} \|\nabla f(\mathbf{w}_t)\|_2 \leq \sqrt{\frac{2[f(\mathbf{w}_0) - f_*]}{\alpha\beta LT}}.$$

Can tune α and β to optimize the bound: since $\alpha + \beta \leq \frac{2}{L}$, the minimum is achieved when $\alpha = \beta = \frac{1}{L}$, and the bound reduces to

$$\min_{0 \leq t \leq T-1} \|\nabla f(\mathbf{w}_t)\|_2 \leq \sqrt{\frac{2L[f(\mathbf{w}_0) - f_*]}{T}},$$

Remarkable Properties

- Rate of convergence is proportional to the Lipschitz smoothness L : the bigger L is, the smaller the step size $\eta = \frac{1}{L}$ has to be since the function f becomes steeper.
- If we start from some point $\mathbf{w}_0$ whose function value is closer to the infimum f_* , then the gradient diminishes faster to zero.
- Very importantly, the rate of convergence does not depend on d , the dimension, at all!
- The $1/\sqrt{T}$ rate of convergence for the gradient is essentially tight¹.

¹C. Cartis, N. I. M. Gould, and P. L. Toint. "On the Complexity of Steepest Descent, Newton's and Regularized Newton's Methods for Nonconvex Unconstrained Optimization". *SIAM Journal on Optimization*, vol. 20, no. 6 (2010), pp. 2833–2852.

Example: Newton's Second Law of Motion

$$\boxed{F = m\mathbf{a}}$$

- Let the force F be given by a potential (energy) U :

$$m\ddot{\mathbf{w}} = -\nabla U(\mathbf{w})$$

- Kinetic energy $K := \frac{1}{2}m\|\dot{\mathbf{w}}\|_2^2$
- Total energy $E = K + U$ is preserved:

$$\dot{E} = m\langle \dot{\mathbf{w}}, \ddot{\mathbf{w}} \rangle + \langle \nabla U(\mathbf{w}), \dot{\mathbf{w}} \rangle = 0$$

- Introduce the momentum $\mathbf{p} := m\dot{\mathbf{w}}$, we have the Hamiltonian

$$H(\mathbf{w}, \mathbf{p}) := \frac{1}{2m}\|\mathbf{p}\|_2^2 + U(\mathbf{w})$$

Gradient Flow, Gradient Descent and Proximal Point

- Gradient flow driven by a smooth or convex function f :

$$\dot{\mathbf{w}}_t = -\nabla f(\mathbf{w}_t)$$

- Forward Euler discretization is gradient descent (Cauchy, 1847):

$$\mathbf{w}_{t+1} = \mathbf{w}_t - \eta_t \cdot \nabla f(\mathbf{w}_t)$$

- for large problems, replace ∇f with a stochastic estimate (Robbins & Monroe, 1951)
- Backward Euler discretization is proximal point (Martinet, 1970; Rockafellar, 1976):

$$\mathbf{w}_{t+1} = \mathbf{w}_t - \eta_t \cdot \nabla f(\mathbf{w}_{t+1}) \quad \text{or} \quad \mathbf{w}_t = \mathbf{w}_{t+1} + \eta_t \cdot \nabla f(\mathbf{w}_{t+1})$$

- not practical, but efficient approximations abound

(Deterministic) Langevin Dynamics

$$m\ddot{\mathbf{w}}_t + \gamma_t \dot{\mathbf{w}}_t + \nabla f(\mathbf{w}_t) = \mathbf{0}$$

- $m = 0$: reduces to gradient flow
- $\gamma_t = 0$: reduces to Newton mechanics
- f acts as potential (energy)
- $\gamma_t \geq 0$ models (dynamic) friction
- More generally, adding random noise we obtain the celebrated Langevin dynamics:

$$m\ddot{\mathbf{w}}_t + \gamma_t \dot{\mathbf{w}}_t + \nabla f(\mathbf{w}_t) = \sqrt{2} \, \mathrm{d}B_t$$

Heavy Ball

$$\ddot{\mathbf{w}}_t + \gamma \dot{\mathbf{w}}_t + \nabla f(\mathbf{w}_t) = \mathbf{0}$$

- Perform discretization: $\frac{\frac{\mathbf{w}_{t+1} - \mathbf{w}_t}{\sqrt{\alpha_t}} - \frac{\mathbf{w}_t - \mathbf{w}_{t-1}}{\sqrt{\alpha_t}}}{\sqrt{\alpha_t}} + \gamma \frac{\mathbf{w}_t - \mathbf{w}_{t-1}}{\sqrt{\alpha_t}} + \nabla f(\mathbf{w}_t) = \mathbf{0}$
- Polyak's heavy ball:

$$\mathbf{w}_{t+1} = \underbrace{\mathbf{w}_t - \alpha_t \nabla f(\mathbf{w}_t)}_{\text{gradient step}} + \underbrace{\beta_t (\mathbf{w}_t - \mathbf{w}_{t-1})}_{\text{momentum}} = \underbrace{(1 + \beta_t) \mathbf{w}_t - \beta_t \mathbf{w}_{t-1}}_{\text{extrapolation}} - \alpha_t \nabla f(\mathbf{w}_t)$$

– $\beta_t = \gamma \sqrt{\alpha_t} - 1 \geq -1$: extrapolation ($\beta_t > 0$) vs. interpolation ($-1 < \beta_t < 0$)

Gradient Averaging

- With $\mathbf{w}_1 = \mathbf{w}_0$ and for all $t \geq 1$:

$$\mathbf{w}_{t+1} - \mathbf{w}_t = \beta_t(\mathbf{w}_t - \mathbf{w}_{t-1}) - \alpha_t \nabla f(\mathbf{w}_t) = - \sum_{k=1}^t \alpha_k \nabla f(\mathbf{w}_k) \prod_{s=k+1}^t \beta_s =: -\eta_t \mathbf{p}_t$$

$$\begin{aligned} \eta_t \mathbf{p}_t &:= \sum_{k=1}^t \alpha_k \nabla f(\mathbf{w}_k) \prod_{s=k+1}^t \beta_s = \alpha_t \nabla f(\mathbf{w}_t) + \sum_{k=1}^{t-1} \alpha_k \nabla f(\mathbf{w}_k) \prod_{s=k+1}^t \beta_s \\ &= \alpha_t \nabla f(\mathbf{w}_t) + \beta_t \eta_{t-1} \mathbf{p}_{t-1} \end{aligned}$$

- Upon choosing $\eta_t = \alpha_t + \beta_t \eta_{t-1}$:

$$\boxed{\mathbf{p}_t = (1 - \mu_t) \mathbf{p}_{t-1} + \mu_t \nabla f(\mathbf{w}_t)}, \quad \text{where} \quad \mu_t := \alpha_t / \eta_t \text{ and } \mu_1 = 1$$

$$\mathbf{w}_{t+1} = \mathbf{w}_t - \eta_t \mathbf{p}_t$$

Nesterov's Momentum

$$\ddot{\mathbf{w}}_t + \frac{\gamma}{t} \dot{\mathbf{w}}_t + \nabla f(\mathbf{w}_t) = \mathbf{0}$$

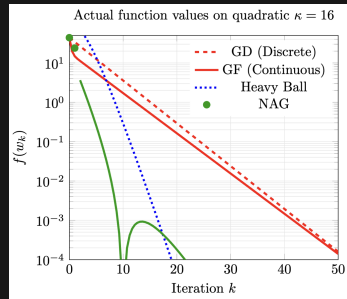
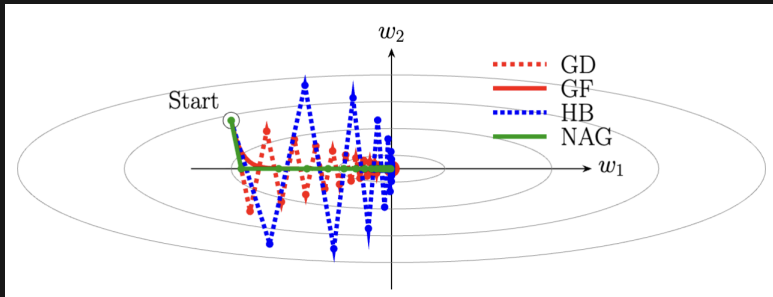
- Diminishing friction $\gamma_t = \frac{\gamma}{t}$ (as opposed to constant in Polyak's original heavy ball)
- $\gamma = 3$ is the critical value
- As before, perform discretization and telescope:

$$\mathbf{z}_t := \mathbf{w}_t + \beta_t(\mathbf{w}_t - \mathbf{w}_{t-1}) = \boxed{\mathbf{w}_t - \eta_t(1 - \mu_t)\mathbf{p}_{t-1}}$$

$$\mathbf{p}_t = (1 - \mu_t)\mathbf{p}_{t-1} + \mu_t \nabla f(\mathbf{z}_t)$$

$$\mathbf{w}_{t+1} = \mathbf{w}_t - \eta_t \mathbf{p}_t$$

– only difference is that gradients are evaluated at the “look-ahead” iterate $\mathbf{z}_t$



generated by Deepseek

Where Is the Noise From?

- Measurement error
- Numerical error
- Scale constraint: most ML problems minimize an averaged loss

$$f(\mathbf{w}) = \frac{1}{n} \sum_{i=1}^n \ell_i(\mathbf{w}), \quad \partial f(\mathbf{w}) = \frac{1}{n} \sum_{i=1}^n \partial \ell_i(\mathbf{w})$$

- Convenience: objective can be reformulated as an expectation

$$f(\mathbf{w}) := \mathbb{E}_{\xi}[f(\mathbf{w}, \xi)], \quad \partial f(\mathbf{w}) = \mathbb{E}_{\xi}[\partial_{\mathbf{w}} f(\mathbf{w}, \xi)]$$

- Regularization: adding noise during training is common in ML
- Privacy: corrupt gradient with noise so that no one can infer user data

Algorithm 3: Stochastic Gradient

Input: $\mathbf{w}_0 \in \text{dom } f$

```
1 for  $t = 0, 1, 2, \dots$  do
2   choose step size  $\eta_t$ 
3   compute stochastic gradient  $\mathbf{g}_t \leftarrow \nabla f(\mathbf{w}_t, \boldsymbol{\xi}_t)$ 
4    $\mathbf{w}_{t+1} \leftarrow \mathbf{P}_C(\mathbf{w}_t - \eta_t \mathbf{g}_t)$ 
5    $\mathbf{z}_t \leftarrow \sum_{k=0}^t \bar{\eta}_{t,k} \mathbf{w}_k$            // ergodic averaging,  $\bar{\eta}_{t,k} := \eta_k / H_t$ ,  $H_t := \sum_{k=0}^t \eta_k$ 
```

- For simplicity, assume stochastic gradient is unbiased, i.e.

$$\mathbb{E}_{\boldsymbol{\xi}_t}[\nabla f(\mathbf{w}_t, \boldsymbol{\xi}_t)] = \nabla f(\mathbf{w}_t)$$

- In general, step size $\eta_t \rightarrow 0$ (or $\sum_t \eta_t^2 < \infty$) and $\sum_t \eta_t = \infty$

Necessity of Diminishing Step Size

- Suppose we are at the minimizer $\mathbf{w}_\star$
- Gradient vanishes at $\nabla f(\mathbf{w}_\star)$
- But stochastic gradient $\nabla f(\mathbf{w}_\star, \boldsymbol{\xi})$ need not be zero
- With a non-vanishing step size, we will wander around $\mathbf{w}_\star$

Computing the Mean

$$f(\mathbf{w}) = \frac{1}{2} \mathbb{E}_{\mathbf{x}} \|\mathbf{w} - \mathbf{x}\|_2^2, \quad \ell_{\mathbf{x}} := \frac{1}{2} \|\mathbf{w} - \mathbf{x}\|_2^2$$

- Obviously, $\mathbf{w}_{\star} = \mathbb{E}[\mathbf{x}]$ is the mean, with $f_{\star} = \frac{1}{2} \mathbb{E}_{\mathbf{x}} \|\mathbf{x} - \mathbb{E}(\mathbf{x})\|_2^2$
- With stochastic gradient:
 - sample $\mathbf{x}_t$
 - compute $\mathbf{w}_{t+1} = \mathbf{w}_t - \eta_t(\mathbf{w}_t - \mathbf{x}_t) = (1 - \eta_t)\mathbf{w}_t + \eta_t\mathbf{x}_t$
 - if $\mathbf{w}_0 = \mathbf{0}$ and $\eta_t = \frac{1}{t+1}$, then $\mathbf{w}_{t+1} = \frac{1}{t+1} \sum_{s=0}^t \mathbf{x}_s$
 - clearly, $\mathbf{w}_t \rightarrow \mathbb{E}[\mathbf{x}]$ and $\mathbb{E}f(\mathbf{w}_t) = (1 + \frac{1}{t})f_{\star}$
 - known to be statistically optimal for $d \leq 2$

Minibatching

- If averaging the stochastic gradient over a minibatch of size b :

$$\mathbf{g} = \frac{1}{b} \sum_{k=1}^b \nabla f(\mathbf{w}, \boldsymbol{\xi}_{i_k})$$

- increase computation by a factor of b
- decrease variance by a factor of b too
- suitable for parallel computation

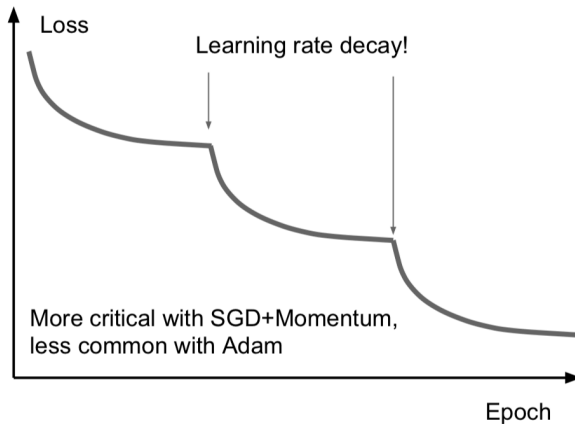
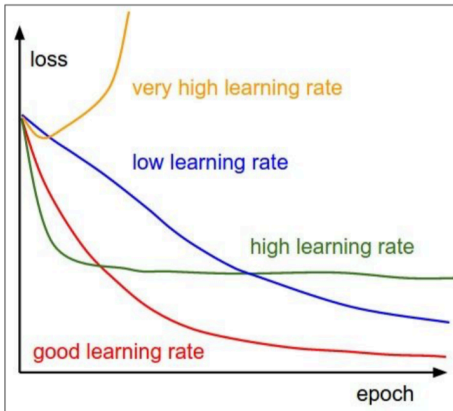
Who is Adam?

$$\mathbf{w}_{t+1} = \mathbf{w}_t - \frac{\eta_t}{\sqrt{\mathbf{s}_t + \epsilon}} \odot \mathbf{v}_t$$

	$\mathbf{s}_t$	$\mathbf{v}_t$
Adagrad	$\mathbf{s}_{t-1} + \nabla g(\mathbf{w}_t) \odot \nabla g(\mathbf{w}_t)$	$\nabla g(\mathbf{w}_t)$
RMSprop	$(1 - \lambda_t)\mathbf{s}_{t-1} + \lambda_t \nabla g(\mathbf{w}_t) \odot \nabla g(\mathbf{w}_t)$	$\nabla g(\mathbf{w}_t)$
Adam	$(1 - \lambda_t)\mathbf{s}_{t-1} + \lambda_t \nabla g(\mathbf{w}_t) \odot \nabla g(\mathbf{w}_t)$	$(1 - \mu_t)\mathbf{v}_{t-1} + \mu_t \nabla g(\mathbf{w}_t)$

bias correction: divide by $\sum_{k=1}^t \lambda_k \prod_{l>k} (1 - \lambda_l)$, similar for μ_t [to retain constant grad]

J. Duchi, E. Hazan, and Y. Singer. "Adaptive Subgradient Methods for Online Learning and Stochastic Optimization". *Journal of Machine Learning Research*, vol. 12, no. 61 (2011), pp. 2121–2159, D. P. Kingma and J. Ba. "Adam: A method for stochastic optimization". In: *3rd International Conference for Learning Representations*. 2015.



- Decrease every few epochs: $\eta_t = \eta_0 \exp(-c \lceil t/k \rceil)$
- Sublinear decay: $\eta_t = \eta_0 / (1 + ct)$ or $\eta_t = \eta_0 / \sqrt{1 + ct}$

MomentUm Orthogonalized by Newton-Schulz

Algorithm 4: Muon

Input: W_0 , stepsize η_t and momentum size μ_t

```
1  $P_{-1} \leftarrow \mathbf{0}$  // initialization
2 for  $t = 0, 1, 2, \dots$  do
3   if Nesterov then
4      $Z_t \leftarrow W_t - \eta_t(1 - \mu_t)P_{t-1}$  // Nesterov momentum
5   else
6      $Z_t \leftarrow W_t$  // Polyak momentum
7    $G_t \leftarrow \nabla \ell(Z_t)$  // compute the gradient
8    $P_t \leftarrow (1 - \mu_t) \cdot P_{t-1} + \mu_t G_t$  // momentum as average of gradients
9    $P_t \leftarrow P_t / \|P_t\|$  // the norm is spectral, or simply Frobenius
10   $O_t \leftarrow \text{NEWTONSCHULZ5}(P_t)$  // orthogonalize by Newton-Schulz
11   $W_{t+1} \leftarrow W_t - \eta_t \cdot O_t$  // update
```

