Adaptive Inertia: Disentangling the Effects of Adaptive Learning Rate and Momentum

Zeke Xie[1,2], Xinrui Wang[1], Huishuai Zhang[3] , Issei Sato[1], and Masashi Sugiyama[2,1]

[1]The University of Tokyo
[2]RIKEN Center for AIP
[3]Microsoft Research Asia

ICML 2022, Long Presentation

東京大学 THE UNIVERSITY OF TOKYO AIP Center for Advanced Intelligence Project

The Mission: Towards the Science of AI

Nowadays deep learning is like physics in/before the time of Galileo.

1. People empirically observed many interesting things.
2. No mathematical theory for most things.

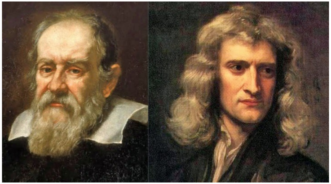


Figure: From the time of Galileo to the time of Newton.

I hope to find a way towards the time of Newton for AI.

1. Science not only explains what works but also predicts what will work.
2. Science gives quantitative and trustworthy results.
3. Science constructs complex principles from first principles.

Science of Deep Learning Dynamics

(Xie et al., ICLR 2021): proposed a physics-inspired diffusion theory for SGD dynamics.

Zeke Xie, Issei Sato, and Masashi Sugiyama. A Diffusion Theory For Deep Learning Dynamics: Stochastic Gradient Descent Exponentially Favors Flat Minima. ICLR2021.

Along this approach, we further analyze why Adam often **converges faster but generalizes worse** than SGD in this work.

- Theory for Momentum and Adam dynamics.
  Adam can escape saddle points efficiently, but cannot favor flat minima as well as SGD.
- New Optimizer: Adaptive Inertia Optimizer (Adai).
  Adai can escape saddle points efficiently like Adam and select flat minima like SGD.

Diffusion Theory for SGD Dynamics

- SGD as continuous-time Langevin Dynamics:

$$d\theta = -\frac{\partial L(\theta)}{\partial \theta} dt + [2D(\theta)]^{\frac{1}{2}} dW_t, \tag{1}$$

where $dW_t \sim \mathcal{N}(0, Idt)$ is a Wiener process and $D(\theta)$ is the diffusion matrix.

- The associated Fokker-Planck Equation:

$$\frac{\partial P(\theta, t)}{\partial t} = \nabla \cdot [P(\theta, t)\nabla L(\theta)] + \nabla \cdot \nabla D(\theta) P(\theta, t) \tag{2}$$

- The dynamics of $\theta \rightarrow$ the diffusion of probability density $P(\theta, t)$

- A physical example: Brownian motion of zero-inertia particles.
- Q: Why Langevin Dynamics?
  A: Predicting θ is intractable, while predicting the distribution of θ is tractable by Langevin Dynamics.

Momentum and Adam

Momentum, known as SGD Momentum or Heavy Ball (Zavriev et al., 1993), uses moving average of past gradients for training.

- A dynamical perspective
  -SGD: a zero-inertia particle.
  -Momentum: a finite-inertia particle.

Algorithm 1: Momentum

$g_t = \nabla L(\theta_{t-1});$

$m_t = \beta_1 m_{t-1} + \beta_3 g_t;$

$\theta_t = \theta_{t-1} - \eta m_t;$

Adam $=$ Momentum $+$ Adaptive Learning Rate.

Adam combines:

1. Momentum: finite inertia
2. Adaptive Learning Rate: anisotropic step sizes (time unit)

Algorithm 2: Adam

$g_t = \nabla L(\theta_{t-1});$

$m_t = \beta_1 m_{t-1} + (1-\beta_1) g_t;$

$v_t = \beta_2 v_{t-1} + (1-\beta_2) g_t^2;$

$\hat{m}_t = \frac{m_t}{1-\beta_1^t};$

$\hat{v}_t = \frac{v_t}{1-\beta_2^t};$

$\theta_t = \theta_{t-1} - \frac{\eta}{\sqrt{\hat{v}_t}+\epsilon} \hat{m}_t;$

The Fokker-Planck Equation for Adam

Inspired by the Newtonian Motion Equation with finite inertia and damping, we obtain the finite-inertia Langevin Dynamics

$$Mdr = -\gamma M d\theta - \frac{\partial L(\theta)}{\partial \theta} dt + [2D]^{\frac{1}{2}} dW_t. \tag{3}$$

$\Longleftrightarrow$ the phase-space Fokker-Planck Equation (the θ-r space) as

$$\begin{aligned} \frac{\partial P(\theta, r, t)}{\partial t} = & - \nabla_\theta \cdot [rP(\theta, r, t)] + \\ & \nabla_r \cdot \left[\gamma r + M^{-1} \nabla_\theta L(\theta)\right] P(\theta, r, t) + \\ & \nabla_r \cdot M^{-2} D \cdot \nabla_r P(\theta, r, t) \end{aligned} \tag{4}$$

where the mass $M = \frac{\eta}{\beta_3}$ and the damping coefficient $\gamma = \frac{1-\beta_1}{\eta}$ (which are all decided by the hyperparameters of deep learning).

Understanding Adam Dynamics

Question: Why does Adam often converge faster but generalize worse than SGD?

Answer: Adam can escape saddle points efficiently, but cannot favor flat minima as well as SGD.

We focus on

- Saddle-point escaping $\Longleftrightarrow$ Convergence speed.
- Minima selection $\Longleftrightarrow$ Generalization.

Escape Saddle Points

- Saddle-point escaping, particularly along very flat directions.
  - Problem Setting: we consider a particle escaping from saddle points.
  - How does the mean squared displacement after certain iterations depend on the Hessian?

How to escape saddle points where gradients are small?

1. Langevin Diffusion helps escape saddle points.
   - The diffusion effect: noise matters.
2. The momentum inertia helps escape saddle points.
   - The drift effect: momentum matters.

Escape Saddle Points: Adam>Momentum>SGD

- SGD: the diffusion effect only.

$$\langle \Delta\theta_i^2 \rangle = \frac{|H_i|\eta^2 T}{B} + \mathcal{O}(B^{-1} H_i^2 \eta^3 T^2),$$

where $\langle \Delta\theta_i^2 \rangle$ is the mean squared displacement and T is the number of iterations.

- Momentum: the diffusion effect and the drift effect.

$$\langle \Delta\theta_i^2 \rangle = \frac{|H_i|\beta_3^2\eta^2}{2(1-\beta_1)^3 B}\left[1 - \exp\left(-(1-\beta_1)T\right)\right]^2 + \frac{|H_i|\beta_3^2\eta^2 T}{B(1-\beta_1)^2} + \mathcal{O}(B^{-1} H_i^2 \eta^3 T^2).$$

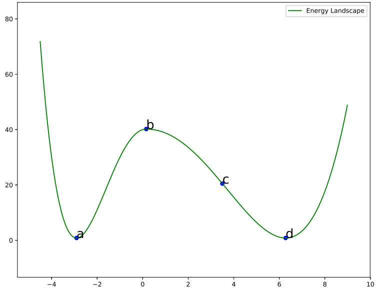
- Adam: the diffusion effect and the drift effect, which are **Hessian-independent**.

$$\langle \Delta\theta_i^2 \rangle = \frac{\eta^2}{2(1-\beta_1)}\left[1 - \exp\left(-(1-\beta_1)T\right)\right]^2 + \eta^2 T + \mathcal{O}(\sqrt{B|H_i|}\eta^3 T^2).$$

Minima Selection as a Kramers Escape Problem

How to describe the escape process from a valley?

- (Kramers, 1940): the diffusion model of chemical reactions
- The escape rate corresponds to the chemical reaction rate.



- The escape rate corresponds to the minima transition rate.
- How many iterations does it take to escape the given valley?
  - SGD is good at escaping sharp minima, while Adam is not.

Select Flat Minima: Momentum≈ SGD > Adam

- SGD generalizes well. (Xie et al., ICLR 2021)

$$\log(\tau) = \mathcal{O}\left(\frac{2B\Delta L}{\eta H_{ae}}\right)$$

where τ is the mean escape time, ΔL is the loss barrier,

and H_{ae} is the minima Hessian eigenvalue along the escape direction.

- Momentum matters little to the mean escape time. Thus, Momentum generalizes well.

$$\log(\tau) = \mathcal{O}\left(\frac{2(1-\beta_1)B\Delta L}{\beta_3 \eta H_{ae}}\right)$$

- Adam cannot escape sharp minima efficiently as SGD. Thus. Adam generalizes worse.

$$\log(\tau) = \mathcal{O}\left(\frac{2\sqrt{B}\Delta L}{\eta \sqrt{H_{ae}}}\right)$$

Adaptive Inertia Optimizer (Adai)

- May we design better optimizers that escape saddle points efficiently and select flat minima well?

- Adaptive Inertia uses adaptive momentum hyperapermeters for different directions.
- Large inertia along flat directions → large drift effects

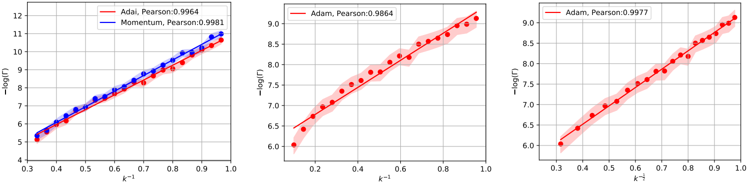
Algorithm 3: Adai

$g_t = \nabla L(\theta_{t-1})$;
$v_t = \beta_2 v_{t-1} + (1 - \beta_2) g_t^2$;
$\hat{v}_t = \frac{v_t}{1-\beta_2^t}$;
$\bar{v}_t = mean(\hat{v}_t)$;
$\mu_t = (1 - \frac{\beta_0}{\bar{v}_t} \hat{v}_t).Clip(0, 1 - \epsilon)$;
$m_t = \mu_t m_{t-1} + (1 - \mu_t) g_t$;
$\hat{m}_t = \frac{m_t}{1-\prod_{z=1}^{t} \mu_z}$;
$\theta_t = \theta_{t-1} - \eta \hat{m}_t$;

- Adai can escape saddle points efficiently like Adam and select flat minima well like SGD.

Empirical Analysis: The Mean Escape Time

- Adam generalizes worse than SGD(/Momentum).
  - Adam: $\log(\tau) \sim k^{-\frac{1}{2}}$.
- Adai generalizes well.
  - Adai/Momentum: $\log(\tau) \sim k^{-1}$.



(a) Adai/SGD: k^{-1} (b) Adam: k^{-1} (c) Adam: $k^{-\frac{1}{2}}$

Figure: Flat Minima Selection: *Adai* $\approx$ *SGD(/Momentum)* $\gg$ *Adam*. Note that k measure the minima sharpness, while the mean escape time τ measures the number of iterations of escaping the given loss valley.

The superiority of Adai

Table: Test performance comparison of optimizers across models and datasets.

Dataset	Model	AdaiW	Adai	SGD M	Adam	AMSGrad	AdamW	AdaBound	Padam	Yogi	RAdam
CIFAR-10	ResNet18	$\mathbf{4.59}_{0.16}$	$4.74_{0.14}$	$5.01_{0.03}$	$6.53_{0.03}$	$6.16_{0.18}$	$5.08_{0.07}$	$5.65_{0.08}$	$5.12_{0.04}$	$5.87_{0.12}$	$6.01_{0.10}$
	VGG16	$\mathbf{5.81}_{0.07}$	$6.00_{0.09}$	$6.42_{0.02}$	$7.31_{0.25}$	$7.14_{0.14}$	$6.48_{0.13}$	$6.76_{0.12}$	$6.15_{0.06}$	$6.90_{0.22}$	$6.56_{0.04}$
CIFAR-100	ResNet34	$21.05_{0.10}$	$\mathbf{20.79}_{0.22}$	$21.52_{0.37}$	$27.16_{0.55}$	$25.53_{0.19}$	$22.99_{0.40}$	$22.87_{0.13}$	$22.72_{0.10}$	$23.57_{0.12}$	$24.41_{0.40}$
	DenseNet121	$\mathbf{19.44}_{0.21}$	$19.59_{0.38}$	$19.81_{0.33}$	$25.11_{0.15}$	$24.43_{0.09}$	$21.55_{0.14}$	$22.69_{0.15}$	$21.10_{0.23}$	$22.15_{0.36}$	$22.27_{0.22}$
	GoogLeNet	$\mathbf{20.50}_{0.25}$	$20.55_{0.32}$	$21.21_{0.29}$	$26.12_{0.33}$	$25.53_{0.17}$	$21.29_{0.17}$	$23.18_{0.31}$	$21.82_{0.17}$	$24.24_{0.16}$	$22.23_{0.15}$

Please refer to our paper for more empirical results.

Summary

1. Adai: A novel adaptive optimization framework, which element-wisely adjust the momentum hyperparameters instead of learning rates.
2. Adai can escape saddle points efficiently like Adam and select flat minima well like SGD.

"Science not only explains what works but also predicts what will work."

Table: Adaptive Learning Rate versus Adaptive Inertia.

	SGD	Adaptive Learning Rate	Adaptive Inertia
Saddle-Escaping	Slow ✖	Fast ✔	Fast ✔
Minima Selection	Flat ✔	Sharp ✖	Flat ✔