Toward Deeper Understanding of Momentum in Nonconvex Stochastic Optimization

Tianyi Liu*, Zhehui Chen*, Enlu Zhou*, Tuo Zhao*
*School of Industrial and System Engineering at Georgia Institute of Technology

Background

Consider an empirical risk minimization problem,

$$\min_{\theta} \mathcal{F}(\theta) := \frac{1}{n}\sum_{i=1}^{n} \ell(y_i, f(x_i, \theta)),$$

- $\{(x_i, y_i)\}_{i=1}^n$: n observations.
- ℓ: loss function.
- f: decision function associated with θ.

A Popular Algorithm: **Momentum SGD (Heavy Ball)**

$$\theta_{k+1} = \theta_k - \eta \nabla \ell(y_i, f(x_i, \theta_k)) \quad \text{SGD}$$
$$+\mu(\theta_k - \theta_{k-1}) \quad \text{Momentum}$$

- η: step size.
- $\mu \in [0, 1)$: momentum parameter.

Questions:

- **Does Momentum accelerate the algorithm?**
- **Does Momentum help escape from saddle?**

Streaming PCA

- A simple, but nontrivial example:

$$\min_{v} \quad -v^\top \mathbb{E}_{x\sim\mathcal{D}}[xx^\top]v$$
$$\text{subject to} \quad v \in \mathcal{M},$$

where $\mathcal{M} = \{v \in \mathbb{R}^p : v^\top v = 1\}$ is the stiefel manifold.

- Streaming data:

  At the i-th iteration, independently sample $x_i \sim \mathcal{D}$.

- Assumption 1 (Eigen-gap):

  $\mathbb{E}[x] = 0$ and $\Sigma = \mathbb{E}[xx^\top]$ is positive definite with eigenvalues

$$\lambda_1 > \lambda_2 \geq ... \geq \lambda_p > 0$$

associated with unit eigenvectors $v^{(1)}$, $v^{(2)}$, ..., $v^{(p)}$.

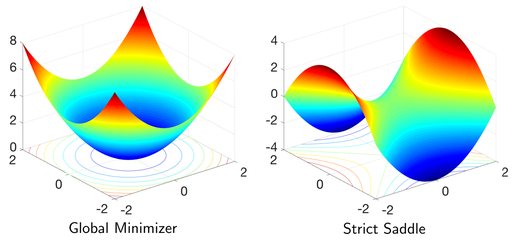
- Assumption 2 (Bounded Input):

$$\|x\|_2 \leq C_p,$$

where C_d is a constant (possibly depends on p).

Landscape

- Stationary point without constraints: $\nabla \mathcal{F}(v) = 0$.



- Stationary point for Streaming PCA: Manifold Gradient:

$$\nabla_{\mathcal{M}} \mathcal{F}(v) = (I - vv^\top)\,\Sigma v = 0.$$

$\Longrightarrow$ **Stationary Points**:

$$\underbrace{\pm v^{(1)}}_{\text{Global Minimizers}}, \quad \underbrace{\pm v^{(2)}, \quad \cdots\cdots, \pm v^{(p)}}_{\text{Strict Saddle and Maximizers}}.$$

Intuition

- MSGD for Streaming PCA:

$$v_{k+1} = v_k + \mu(v_k - v_{k-1}) + \eta \underbrace{(I - v_k v_k^\top) X_k X_k^\top v_k}_{\nabla_{\mathcal{M}} F_k(v_k)\text{: Stoc. Approx. of } \nabla_{\mathcal{M}}\mathcal{F}(v_k).},$$

- ODE Approximation:

$$\text{Discrete:} \quad \frac{v_{k+1} - v_k}{\eta} = \mu\frac{v_k - v_{k-1}}{\eta} + \nabla_{\mathcal{M}} F_k(v_k).$$
$$\text{weakly} \Big\downarrow \ \eta \to 0$$
$$\text{Continuous:} \quad dV = \mu dV + \nabla_{\mathcal{M}}\mathcal{F}(V).$$

Discrete/Stochastic → Continuous/Deterministic.
Similar to the Law of Large Number, not reliable!

- SDE Approximation ($v_k^\eta - v^{(i)} = O(\sqrt{\eta})$):

  Consider the normalized error

$$u_k^\eta = \frac{v_k^\eta - v^{(i)}}{\sqrt{\eta}}.$$

$$\text{Discrete:} \quad \frac{u_{k+1} - u_k}{\sqrt{\eta}} = \mu\frac{u_k - u_{k-1}}{\sqrt{\eta}} + \nabla_{\mathcal{M}} F_k(u_k).$$
$$\text{weakly} \Big\downarrow \ \eta \to 0$$
$$\text{Continuous:} \quad dU = \mu dU + \nabla_{\mathcal{M}}\mathcal{F}(U)dt + \Sigma dB_t.$$

Randomness Returns.
Similar to the Central Limit Theorem!

Diffusion Approximation

Theorem 1 (Global Convergence).
Chosen $\mu \in [0, 1)$, $v_0 \in \mathcal{M}$, denote $V^\eta(t) = v_{\lfloor t/\eta \rfloor}$.

$$V^\eta(\cdot) \xRightarrow[\text{weakly converge}]{\eta\to 0} V(\cdot) \quad \textit{satisfying}$$

ODE $\quad dV = \frac{1}{1-\mu}[\Sigma V - V^\top \Sigma V V]dt, \ V(0) = v_0.$

Theorem 2 (Local Behavior).
Denote $U^\eta(t) = u^\eta_{\lfloor t/\eta \rfloor}$. Based on $v_k^\eta - v^{(i)} = O(\sqrt{\eta})$,

$$\textit{for } i \neq j, \quad U_i^\eta(t) \xRightarrow[\text{weakly converge}]{\eta\to 0} U_i(t) \quad \textit{satisfying}$$

SDE $\quad dU_i = \underbrace{\frac{\lambda_i - \lambda_j}{1-\mu} U_i dt}_{\text{Drift}} + \underbrace{\frac{\alpha_{i,j}}{1-\mu} dB_t}_{\text{Noise}}.$

Proof Technique: Fixed-State-Chain [1].

Remark 3. *O-U processes $\{U_i\}$ characterize the local algorithmic behavior:*

- *$\lambda_i < \lambda_j$, process converges.*
- *$\lambda_i > \lambda_j$, process diverges.*

Reference: [1] Kushner and Yin. Stochastic approximation and recursive algorithms and applications, Stochastic modelling and applied probability, vol. 35.

Implication

- **Phase I (Around Saddle)**: Momentum enlarges variance, encourages exploitation and helps escape!
- **Phase II (Traverse between stationary points)**: Drift dominating variance leads to deterministic trajectory.
- **Phase III (Around Optima)**: Momentum enlarges variance and hurts convergence. Step size annealing!

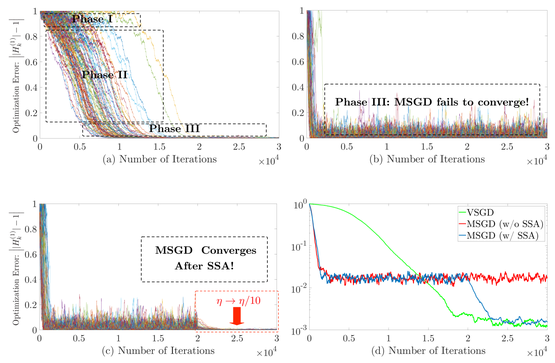
Phase	Step Size η	Asymptotic Complexity
I	$\frac{(\lambda_1-\lambda_2)\epsilon}{\phi}$	$\frac{(1-\mu)\phi}{\epsilon(\lambda_1-\lambda_2)^2}\log\left(\frac{(1-\mu)\delta^2}{\epsilon}\right)$
II	$\frac{(\lambda_1-\lambda_2)\epsilon}{\phi}$	$\frac{(1-\mu)\phi}{\epsilon(\lambda_1-\lambda_2)^2}\log\left(\frac{1-\delta^2}{\delta^2}\right)$
III	$\frac{(1-\mu)(\lambda_1-\lambda_2)\epsilon}{\phi}$	$\frac{\phi}{\epsilon(\lambda_1-\lambda_2)^2}\log\left(\frac{\delta^2}{\epsilon}\right)$

ϕ is a constant related to $\alpha_{i,j}$.
Momentum accelerates the algorithm in Phase I and II.

Experiments

- Streaming PCA (Synthetic Data):

  $\Sigma = \text{diag}\{4, 3, 2, 1\}, \mu = 0.9, \eta = 5 \times 10^{-5}$.



  ♢ Momentum helps escape from saddle.

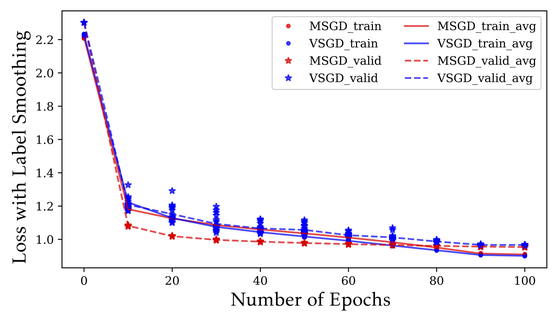
  ♢ MSGD converges after step size annealing.

- Deep Neural Network with optimal setting:

  Architecture: ResNet-9; Dataset: CIFAR-10;

  Peak Step Size: $\eta_{\text{V}} = 2$; $\eta_{\text{M}} = 0.36$ with $\mu = 0.9$;

  Step Size Linear Schedule:
  $0 - 20$ epochs: $0 \nearrow \eta$; $21 - 100$ epochs: $\eta \searrow 0$.



  ♢ The best setting of MSGD escapes from saddle much faster than that of VSGD and eventually outperforms that of VSGD.

Open Questions

1. Extension to more general problems;
2. Bridging asymptotic and nonasymptotic analyses.