Online Learning Guided Quasi-Newton Methods: Improved Global Non-asymptotic Guarantees

Aryan Mokhtari

ECE Department, UT Austin

Based on joint work with Ruichen Jiang and Qiujiang Jin

Optimization for Machine Learning Workshop, NeurIPS 2024
December 15th, 2024

What is the problem of interest?

- Consider the general unconstrained minimization problem

$$\min_{\boldsymbol{x} \in \mathbb{R}^d} \quad f(\boldsymbol{x}),$$

where ∇f is L_1-Lipschitz and $\nabla^2 f$ is L_2-Lipschitz

What is the problem of interest?

- Consider the general unconstrained minimization problem

$$\min_{\boldsymbol{x} \in \mathbb{R}^d} \quad f(\boldsymbol{x}),$$

where ∇f is L_1-Lipschitz and $\nabla^2 f$ is L_2-Lipschitz

- We will focus on two general settings
  - Case I: f is μ-strongly convex (SCVX)
  - Case II: f is (only) convex (CVX)

What is the problem of interest?

- Consider the general unconstrained minimization problem

$$\min_{\boldsymbol{x}\in\mathbb{R}^d} \quad f(\boldsymbol{x}),$$

where ∇f is L_1-Lipschitz and $\nabla^2 f$ is L_2-Lipschitz

- We will focus on two general settings
  - Case I: f is μ-strongly convex (SCVX)
  - Case II: f is (only) convex (CVX)

- We are interested in settings where we can only query first-order information
$\Rightarrow$ We only have access to $\nabla f(x)$

Gradient Descent-type Methods

- Popular methods: Gradient Descent (GD) and its Accelerated version (AGD)
  - Require only access to gradient oracle $\Rightarrow$ Cost per iteration $\mathcal{O}(d)$
  - In Case I (SCVX): Achieve a global linear convergence rate

$$\|\boldsymbol{x}_k - \boldsymbol{x}^*\|^2 \leq \begin{cases} (1-\frac{\mu}{L_1})^k \|\boldsymbol{x}_0 - \boldsymbol{x}^*\|^2 & \text{for GD;} \\ (1-\sqrt{\frac{\mu}{L_1}})^k \|\boldsymbol{x}_0 - \boldsymbol{x}^*\|^2 & \text{for AGD.} \end{cases}$$

  - In Case II (CVX): Achieve a global sublinear convergence rate

$$f(\boldsymbol{x}_k) - f^* = \begin{cases} \mathcal{O}(\frac{1}{k}) & \text{for GD;} \\ \mathcal{O}(\frac{1}{k^2}) & \text{for AGD.} \end{cases}$$

Can we improve over GD/AGD?

- AGD is "worst-case optimal": it matches lower bounds for all first-order methods in both SCVX and CVX settings [Nesterov'18]

Can we improve over GD/AGD?

- AGD is "worst-case optimal": it matches lower bounds for all first-order methods in both SCVX and CVX settings [Nesterov'18]
- However, these lower bounds only hold in the high-dimensional regime, e.g., $k = \mathcal{O}(d)$

Can we improve over GD/AGD?

- AGD is "worst-case optimal": it matches lower bounds for all first-order methods in both SCVX and CVX settings [Nesterov'18]
- However, these lower bounds only hold in the high-dimensional regime, e.g., $k = \mathcal{O}(d)$
- We propose a quasi-Newton-type method that:
  - Matches the rate of GD/AGD when $k = \mathcal{O}(d)$
  - Outperforms GD/AGD with a faster rate when $k = \Omega(d)$

Quasi-Newton Methods

- Quasi-Newton (QN) methods aim at speeding up GD-type methods by using a preconditioner

$$\boldsymbol{x}_{k+1} = \boldsymbol{x}_k - \eta_k \mathbf{B}_k^{-1} \nabla f(\boldsymbol{x}_k)$$

- When $\mathbf{B}_k \approx \nabla^2 f(\boldsymbol{x}_k)$ they mimic Newton's method

Quasi-Newton Methods

- Quasi-Newton (QN) methods aim at speeding up GD-type methods by using a preconditioner
$$\mathbf{x}_{k+1} = \mathbf{x}_k - \eta_k \mathbf{B}_k^{-1} \nabla f(\mathbf{x}_k)$$
- When $\mathbf{B}_k \approx \nabla^2 f(\mathbf{x}_k)$ they mimic Newton's method
- Only use gradients to construct $\mathbf{B}_k \Rightarrow$ Still first-order methods

Quasi-Newton Methods

- Quasi-Newton (QN) methods aim at speeding up GD-type methods by using a preconditioner

$$\boldsymbol{x}_{k+1} = \boldsymbol{x}_k - \eta_k \mathbf{B}_k^{-1} \nabla f(\boldsymbol{x}_k)$$

- When $\mathbf{B}_k \approx \nabla^2 f(\boldsymbol{x}_k)$ they mimic Newton's method
- Only use gradients to construct $\mathbf{B}_k \Rightarrow$ Still first-order methods
- Various updates for $\mathbf{B}_k$ have been proposed with cost $\mathcal{O}(d^2)$: DFP [Davidon'59; Fletcher-Powell'63], BFGS [Broyden'70; Fletcher'70; Goldfarb'70; Shanno'70], SR1 [Powell'69]

Quasi-Newton Methods

- Quasi-Newton (QN) methods aim at speeding up GD-type methods by using a preconditioner
$$\boldsymbol{x}_{k+1} = \boldsymbol{x}_k - \eta_k \mathbf{B}_k^{-1} \nabla f(\boldsymbol{x}_k)$$
- When $\mathbf{B}_k \approx \nabla^2 f(\boldsymbol{x}_k)$ they mimic Newton's method
- Only use gradients to construct $\mathbf{B}_k \Rightarrow$ Still first-order methods
- Various updates for $\mathbf{B}_k$ have been proposed with cost $\mathcal{O}(d^2)$: DFP [Davidon'59; Fletcher-Powell'63], BFGS [Broyden'70; Fletcher'70; Goldfarb'70; Shanno'70], SR1 [Powell'69]
- Despite their practical success, no result shows an improved global complexity bound for QN methods

Prior Work on QN Methods: SCVX setting

- Classical results show asymptotic superlinear convergence, i.e., $\lim_{k\to\infty} \frac{\|\mathbf{x}_{k+1}-\mathbf{x}^*\|}{\|\mathbf{x}_k-\mathbf{x}^*\|} = 0$
  [Powell'71; Broyden-Dennis-Moré'73; Powell'76; ...]
  $\Longrightarrow$ No explicit rates are given

Prior Work on QN Methods: SCVX setting

- Classical results show asymptotic superlinear convergence, i.e., $\lim_{k\to\infty} \frac{\|\mathbf{x}_{k+1}-\mathbf{x}^*\|}{\|\mathbf{x}_k-\mathbf{x}^*\|} = 0$
  [Powell'71; Broyden-Dennis-Moré'73; Powell'76; ...]
  $\Longrightarrow$ No explicit rates are given

- Recent results show a local non-asymptotic superlinear rate of $\left(\mathcal{O}(1/\sqrt{k})\right)^k$
  [Rodomanov-Nesterov'21; Jin-Mokhtari'22; ...]
  - These results are only local. Unclear how to extend them into global guarantees
    $\Rightarrow$ The condition on $\mathbf{B}_0$ may not hold when $\|\mathbf{x}_0-\mathbf{x}^*\|$ becomes small
  - Moreover, there is no global result matching the linear rate of AGD or GD

Prior Work on QN Methods: CVX Setting

- In the CVX setting, few results are known for classical QN methods
  - $\lim_{k\to\infty} f(\boldsymbol{x}_k) = f(\boldsymbol{x}^*)$ with exact line search [Powell'72]
  - $\liminf_{k\to\infty} \|\nabla f(\boldsymbol{x}_k)\| = 0$ with inexact line search [Powell'76; Byrd-Nocedal-Yuan'87]

Prior Work on QN Methods: CVX Setting

- In the CVX setting, few results are known for classical QN methods
  - $\lim_{k\to\infty} f(\mathbf{x}_k) = f(\mathbf{x}^*)$ with exact line search [Powell'72]
  - $\liminf_{k\to\infty} \|\nabla f(\mathbf{x}_k)\| = 0$ with inexact line search [Powell'76; Byrd-Nocedal-Yuan'87]
- Another line of work analyzed QN methods as preconditioned GD methods
  - $\mathcal{O}(1/k)$ rate is shown in [Scheinberg-Tang'16]
  - An accelerated $\mathcal{O}(1/k^2)$ rate is achieved in [Ghanbari-Scheinberg'18]
- However, these rates are no better than that of AGD

Goal and Main Ideas of our Proposed Approach

- **Goal:** Designing QN methods with superior gradient complexity compared to GD-type methods in both CVX and SCVX settings.

Goal and Main Ideas of our Proposed Approach

- **Goal:** Designing QN methods with superior gradient complexity compared to GD-type methods in both CVX and SCVX settings.
- **Our Approach:** Online-Learning guided Quasi-Newton Proximal Extragradient (QNPE) Algorithms

Goal and Main Ideas of our Proposed Approach

- **Goal:** Designing QN methods with superior gradient complexity compared to GD-type methods in both CVX and SCVX settings.
- **Our Approach:** Online-Learning guided Quasi-Newton Proximal Extragradient (QNPE) Algorithms
- **Main Ideas:**
  - Instead of the classic template of QN methods ($\boldsymbol{x}_{k+1} = \boldsymbol{x}_k - \eta_k \mathbf{B}_k^{-1} \nabla f(\boldsymbol{x}_k)$), we follow the Hybrid Proximal Extragradient (HPE) framework
  - Instead of updating $\mathbf{B}_k$ by classic QN updates, we use an Online Learning framework for updating $\mathbf{B}_k$ inspired by our analysis

Our Contributions (Strongly-Convex Setting)

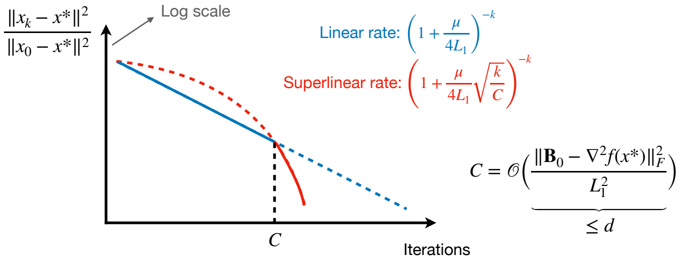
- Global convergence rates (no conditions on $\boldsymbol{x}_0$ or $\mathbf{B}_0$) [Jiang-Jin-M, COLT '23]

$$\frac{\|\boldsymbol{x}_k - \boldsymbol{x}^*\|^2}{\|\boldsymbol{x}_0 - \boldsymbol{x}^*\|^2} \leq \min\left\{\left(1 + \frac{\mu}{4L_1}\right)^{-k}, \left(1 + \frac{\mu}{4L_1}\sqrt{\frac{k}{C}}\right)^{-k}\right\}$$

Our Contributions (Strongly-Convex Setting)

- Global convergence rates (no conditions on $\boldsymbol{x}_0$ or $\mathbf{B}_0$) [Jiang-Jin-M, COLT '23]

$$\frac{\|\boldsymbol{x}_k - \boldsymbol{x}^*\|^2}{\|\boldsymbol{x}_0 - \boldsymbol{x}^*\|^2} \leq \min\left\{\left(1 + \frac{\mu}{4L_1}\right)^{-k}, \left(1 + \frac{\mu}{4L_1}\sqrt{\frac{k}{C}}\right)^{-k}\right\}$$



- For $k \leq d$, QNPE matches the linear rate of GD
- After at most $\mathcal{O}(d)$ iterations QNPE becomes provably faster than GD

Our Contributions (Convex Setting)

- An accelerated QN proximal extragradient method [Jiang-M, NeurIPS '23]

$$f(\boldsymbol{x}_k) - f(\boldsymbol{x}^*) \leq \mathcal{O}\left(\min\left\{\frac{1}{k^2}, \frac{\sqrt{d\log k}}{k^{2.5}}\right\}\right)$$

  - for $k \leq d\log d$, it matches the rate of AGD
  - for $k \geq d\log d$, it provably converges faster than AGD

Our Contributions (Convex Setting)

- An accelerated QN proximal extragradient method [Jiang-M, NeurIPS '23]

$$f(\boldsymbol{x}_k) - f(\boldsymbol{x}^*) \leq \mathcal{O}\left(\min\left\{\frac{1}{k^2}, \frac{\sqrt{d \log k}}{k^{2.5}}\right\}\right)$$

  - for $k \leq d \log d$, it matches the rate of AGD
  - for $k \geq d \log d$, it provably converges faster than AGD

- Lower bound discussion:
  - This result does not violate the lower bound for first-order methods
  - The lower bound of $\Omega\left(\frac{1}{k^2}\right)$ only holds for $k \leq d$

Hybrid Proximal Extragradient

- We follow (a variant of) the Hybrid Proximal Extragradient (HPE) framework [Solodov-Svaiter'99; Monteiro-Svaiter'10]

Hybrid Proximal Extragradient

- We follow (a variant of) the Hybrid Proximal Extragradient (HPE) framework [Solodov-Svaiter'99; Monteiro-Svaiter'10]
- Stage 1: Inexact proximal point update

$$\hat{\boldsymbol{x}}_k \approx \boldsymbol{x}_k - \eta_k \nabla f(\hat{\boldsymbol{x}}_k)$$

Hybrid Proximal Extragradient

- We follow (a variant of) the Hybrid Proximal Extragradient (HPE) framework [Solodov-Svaiter'99; Monteiro-Svaiter'10]
- Stage 1: Inexact proximal point update

$$\|\hat{\boldsymbol{x}}_k - \boldsymbol{x}_k + \eta_k \nabla f(\hat{\boldsymbol{x}}_k)\| \leq \frac{1}{2}\|\hat{\boldsymbol{x}}_k - \boldsymbol{x}_k\|$$

Hybrid Proximal Extragradient

- We follow (a variant of) the Hybrid Proximal Extragradient (HPE) framework [Solodov-Svaiter'99; Monteiro-Svaiter'10]
- Stage 1: Inexact proximal point update

$$\|\hat{\boldsymbol{x}}_k - \boldsymbol{x}_k + \eta_k \nabla f(\hat{\boldsymbol{x}}_k)\| \leq \frac{1}{2}\|\hat{\boldsymbol{x}}_k - \boldsymbol{x}_k\|$$

- Stage 2: Extragradient step

$$\boldsymbol{x}_{k+1} = \gamma_k[\boldsymbol{x}_k - \eta_k \nabla f(\hat{\boldsymbol{x}}_k)] + (1-\gamma_k)\hat{\boldsymbol{x}}_k, \quad \gamma_k = \frac{1}{1+2\eta_k\mu}$$

Hybrid Proximal Extragradient

- We follow (a variant of) the Hybrid Proximal Extragradient (HPE) framework [Solodov-Svaiter'99; Monteiro-Svaiter'10]
- Stage 1: Inexact proximal point update

$$\|\hat{\boldsymbol{x}}_k - \boldsymbol{x}_k + \eta_k \nabla f(\hat{\boldsymbol{x}}_k)\| \leq \frac{1}{2}\|\hat{\boldsymbol{x}}_k - \boldsymbol{x}_k\|$$

- Stage 2: Extragradient step

$$\boldsymbol{x}_{k+1} = \gamma_k[\boldsymbol{x}_k - \eta_k \nabla f(\hat{\boldsymbol{x}}_k)] + (1-\gamma_k)\hat{\boldsymbol{x}}_k, \quad \gamma_k = \frac{1}{1+2\eta_k\mu}$$

Hybrid Proximal Extragradient

- We follow (a variant of) the Hybrid Proximal Extragradient (HPE) framework [Solodov-Svaiter'99; Monteiro-Svaiter'10]
- Stage 1: Inexact proximal point update

$$\|\hat{\boldsymbol{x}}_k - \boldsymbol{x}_k + \eta_k \nabla f(\hat{\boldsymbol{x}}_k)\| \leq \frac{1}{2}\|\hat{\boldsymbol{x}}_k - \boldsymbol{x}_k\|$$

- Stage 2: Extragradient step

$$\boldsymbol{x}_{k+1} = \gamma_k[\boldsymbol{x}_k - \eta_k \nabla f(\hat{\boldsymbol{x}}_k)] + (1-\gamma_k)\hat{\boldsymbol{x}}_k, \quad \gamma_k = \frac{1}{1+2\eta_k\mu}$$

- $\|\boldsymbol{x}_{k+1} - \boldsymbol{x}^*\|^2 \leq \frac{1}{1+2\eta_k\mu}\|\boldsymbol{x}_k - \boldsymbol{x}^*\|^2 \Rightarrow$ any rate can be achieved as $\eta_k \uparrow$

Quasi-Newton Proximal Extragradient

- Issue: Subproblem in Stage 1 is costly!

$$\|\hat{\boldsymbol{x}}_k - \boldsymbol{x}_k + \eta_k \nabla f(\hat{\boldsymbol{x}}_k)\| \leq \frac{1}{2}\|\hat{\boldsymbol{x}}_k - \boldsymbol{x}_k\|$$

Quasi-Newton Proximal Extragradient

- Issue: Subproblem in Stage 1 is costly!

$$\|\hat{\mathbf{x}}_k - \mathbf{x}_k + \eta_k \nabla f(\hat{\mathbf{x}}_k)\| \leq \frac{1}{2}\|\hat{\mathbf{x}}_k - \mathbf{x}_k\|$$

- Solution: Linearize $\nabla f(\hat{\mathbf{x}}_k) \approx \nabla f(\mathbf{x}_k) + \mathbf{B}_k(\hat{\mathbf{x}}_k - \mathbf{x}_k) \Rightarrow$ a linear system of equations

Quasi-Newton Proximal Extragradient

- Issue: Subproblem in Stage 1 is costly!

$$\|\hat{\mathbf{x}}_k - \mathbf{x}_k + \eta_k \nabla f(\hat{\mathbf{x}}_k)\| \leq \frac{1}{2}\|\hat{\mathbf{x}}_k - \mathbf{x}_k\|$$

- Solution: Linearize $\nabla f(\hat{\mathbf{x}}_k) \approx \nabla f(\mathbf{x}_k) + \mathbf{B}_k(\hat{\mathbf{x}}_k - \mathbf{x}_k) \Rightarrow$ a linear system of equations
- Stage 1: Quasi-Newton proximal step

$$\|\hat{\mathbf{x}}_k - \mathbf{x}_k + \eta_k(\nabla f(\mathbf{x}_k) + \mathbf{B}_k(\hat{\mathbf{x}}_k - \mathbf{x}_k))\| \leq \frac{1}{4}\|\hat{\mathbf{x}}_k - \mathbf{x}_k\|,$$

$$\eta_k\|\nabla f(\hat{\mathbf{x}}_k) - (\nabla f(\mathbf{x}_k) + \mathbf{B}_k(\hat{\mathbf{x}}_k - \mathbf{x}_k))\| \leq \frac{1}{4}\|\hat{\mathbf{x}}_k - \mathbf{x}_k\|$$

Quasi-Newton Proximal Extragradient

- Issue: Subproblem in Stage 1 is costly!

$$\|\hat{\boldsymbol{x}}_k - \boldsymbol{x}_k + \eta_k \nabla f(\hat{\boldsymbol{x}}_k)\| \leq \frac{1}{2}\|\hat{\boldsymbol{x}}_k - \boldsymbol{x}_k\|$$

- Solution: Linearize $\nabla f(\hat{\boldsymbol{x}}_k) \approx \nabla f(\boldsymbol{x}_k) + \mathbf{B}_k(\hat{\boldsymbol{x}}_k - \boldsymbol{x}_k) \Rightarrow$ a linear system of equations
- Stage 1: Quasi-Newton proximal step

$$\|\hat{\boldsymbol{x}}_k - \boldsymbol{x}_k + \eta_k(\nabla f(\boldsymbol{x}_k) + \mathbf{B}_k(\hat{\boldsymbol{x}}_k - \boldsymbol{x}_k))\| \leq \frac{1}{4}\|\hat{\boldsymbol{x}}_k - \boldsymbol{x}_k\|,$$

$$\eta_k\|\nabla f(\hat{\boldsymbol{x}}_k) - (\nabla f(\boldsymbol{x}_k) + \mathbf{B}_k(\hat{\boldsymbol{x}}_k - \boldsymbol{x}_k))\| \leq \frac{1}{4}\|\hat{\boldsymbol{x}}_k - \boldsymbol{x}_k\|$$

Quasi-Newton Proximal Extragradient

- Issue: Subproblem in Stage 1 is costly!

$$\|\hat{\mathbf{x}}_k - \mathbf{x}_k + \eta_k \nabla f(\hat{\mathbf{x}}_k)\| \leq \frac{1}{2}\|\hat{\mathbf{x}}_k - \mathbf{x}_k\|$$

- Solution: Linearize $\nabla f(\hat{\mathbf{x}}_k) \approx \nabla f(\mathbf{x}_k) + \mathbf{B}_k(\hat{\mathbf{x}}_k - \mathbf{x}_k) \Rightarrow$ a linear system of equations
- Stage 1: Quasi-Newton proximal step

$$\|\hat{\mathbf{x}}_k - \mathbf{x}_k + \eta_k(\nabla f(\mathbf{x}_k) + \mathbf{B}_k(\hat{\mathbf{x}}_k - \mathbf{x}_k))\| \leq \frac{1}{4}\|\hat{\mathbf{x}}_k - \mathbf{x}_k\|,$$

$$\eta_k\|\nabla f(\hat{\mathbf{x}}_k) - (\nabla f(\mathbf{x}_k) + \mathbf{B}_k(\hat{\mathbf{x}}_k - \mathbf{x}_k))\| \leq \frac{1}{4}\|\hat{\mathbf{x}}_k - \mathbf{x}_k\|$$

  - η_k is not arbitrary; requires backtracking line search

Quasi-Newton Proximal Extragradient

- Issue: Subproblem in Stage 1 is costly!

$$\|\hat{\mathbf{x}}_k - \mathbf{x}_k + \eta_k \nabla f(\hat{\mathbf{x}}_k)\| \leq \frac{1}{2}\|\hat{\mathbf{x}}_k - \mathbf{x}_k\|$$

- Solution: Linearize $\nabla f(\hat{\mathbf{x}}_k) \approx \nabla f(\mathbf{x}_k) + \mathbf{B}_k(\hat{\mathbf{x}}_k - \mathbf{x}_k) \Rightarrow$ a linear system of equations
- Stage 1: Quasi-Newton proximal step

$$\|\hat{\mathbf{x}}_k - \mathbf{x}_k + \eta_k(\nabla f(\mathbf{x}_k) + \mathbf{B}_k(\hat{\mathbf{x}}_k - \mathbf{x}_k))\| \leq \frac{1}{4}\|\hat{\mathbf{x}}_k - \mathbf{x}_k\|,$$

$$\eta_k\|\nabla f(\hat{\mathbf{x}}_k) - (\nabla f(\mathbf{x}_k) + \mathbf{B}_k(\hat{\mathbf{x}}_k - \mathbf{x}_k))\| \leq \frac{1}{4}\|\hat{\mathbf{x}}_k - \mathbf{x}_k\|$$

  - η_k is not arbitrary; requires backtracking line search
- Stage 2: Extragradient step

$$\mathbf{x}_{k+1} = \gamma_k[\mathbf{x}_k - \eta_k \nabla f(\hat{\mathbf{x}}_k)] + (1 - \gamma_k)\hat{\mathbf{x}}_k, \quad \gamma_k = \frac{1}{1 + 2\eta_k \mu}$$

How to Update $\mathbf{B}_k$: Starting from Convergence Analysis

- How should we select/update $\{\mathbf{B}_k\}$?

How to Update $\mathbf{B}_k$: Starting from Convergence Analysis

- How should we select/update $\{\mathbf{B}_k\}$?
- We let the convergence analysis guide our choice of $\mathbf{B}_k$!

How to Update $\mathbf{B}_k$: Starting from Convergence Analysis

- How should we select/update $\{\mathbf{B}_k\}$?
- We let the convergence analysis guide our choice of $\mathbf{B}_k$!
- We know that $\|\boldsymbol{x}_{k+1} - \boldsymbol{x}^*\|^2 \leq \frac{1}{1+2\eta_k\mu}\|\boldsymbol{x}_k - \boldsymbol{x}^*\|^2$

How to Update $\mathbf{B}_k$: Starting from Convergence Analysis

- How should we select/update $\{\mathbf{B}_k\}$?
- We let the convergence analysis guide our choice of $\mathbf{B}_k$!
- We know that $\|\boldsymbol{x}_{k+1} - \boldsymbol{x}^*\|^2 \leq \frac{1}{1+2\eta_k\mu}\|\boldsymbol{x}_k - \boldsymbol{x}^*\|^2$
- η_k is constrained by

$$\eta_k\|\nabla f(\hat{\boldsymbol{x}}_k) - (\nabla f(\boldsymbol{x}_k) + \mathbf{B}_k(\hat{\boldsymbol{x}}_k - \boldsymbol{x}_k))\| \leq \frac{1}{4}\|\hat{\boldsymbol{x}}_k - \boldsymbol{x}_k\|$$

- **Initial result:** By backtracking line search:

$$\eta_k \simeq \frac{\|\hat{\boldsymbol{x}}_k - \boldsymbol{x}_k\|}{\|\nabla f(\hat{\boldsymbol{x}}_k) - \nabla f(\boldsymbol{x}_k) - \mathbf{B}_k(\hat{\boldsymbol{x}}_k - \boldsymbol{x}_k)\|} = \frac{\|\boldsymbol{s}_k\|}{\|\boldsymbol{y}_k - \mathbf{B}_k\boldsymbol{s}_k\|},$$

where $\boldsymbol{y}_k = \nabla f(\hat{\boldsymbol{x}}_k) - \nabla f(\boldsymbol{x}_k)$ and $\boldsymbol{s}_k = \hat{\boldsymbol{x}}_k - \boldsymbol{x}_k$

How to Update $\mathbf{B}_k$: Starting from Convergence Analysis

- Since $\eta_k \simeq \|\boldsymbol{s}_k\|/\|\boldsymbol{y}_k - \mathbf{B}_k \boldsymbol{s}_k\|$, by applying Jensen's inequality, we have

$$\frac{\|\boldsymbol{x}_N - \boldsymbol{x}^*\|^2}{\|\boldsymbol{x}_0 - \boldsymbol{x}^*\|^2} \leq \left(1 + 2\mu\sqrt{\frac{N}{\mathcal{O}(\sum_{k=0}^{N-1} \frac{\|\boldsymbol{y}_k - \mathbf{B}_k \boldsymbol{s}_k\|^2}{\|\boldsymbol{s}_k\|^2})}}\right)^{-N}$$

How to Update $\mathbf{B}_k$: Starting from Convergence Analysis

- Since $\eta_k \simeq \|\boldsymbol{s}_k\|/\|\boldsymbol{y}_k - \mathbf{B}_k \boldsymbol{s}_k\|$, by applying Jensen's inequality, we have

$$\frac{\|\boldsymbol{x}_N - \boldsymbol{x}^*\|^2}{\|\boldsymbol{x}_0 - \boldsymbol{x}^*\|^2} \leq \left(1 + 2\mu \sqrt{\frac{N}{\mathcal{O}(\sum_{k=0}^{N-1} \frac{\|\boldsymbol{y}_k - \mathbf{B}_k \boldsymbol{s}_k\|^2}{\|\boldsymbol{s}_k\|^2})}}\right)^{-N}$$

- Hence, the goal is to choose $\mathbf{B}_k$ such that we minimize

$$\sum_{k=0}^{N-1} \ell_k(\mathbf{B}_k) \; := \; \sum_{k=0}^{N-1} \frac{\|\boldsymbol{y}_k - \mathbf{B}_k \boldsymbol{s}_k\|^2}{\|\boldsymbol{s}_k\|^2}$$

Hessian Approximation Update via Online Learning

- We aim to minimize $\sum_{k=0}^{N-1} \ell_k(\mathbf{B}_k)$, but we observe ℓ_k after selecting $\mathbf{B}_k$!

Hessian Approximation Update via Online Learning

- We aim to minimize $\sum_{k=0}^{N-1} \ell_k(\mathbf{B}_k)$, but we observe ℓ_k after selecting $\mathbf{B}_k$!

  Why? given $\boldsymbol{x}_k, \mathbf{B}_k \rightarrow$ select $(\eta_k, \hat{\boldsymbol{x}}_k)$ by BLS $\rightarrow$ compute $\boldsymbol{s}_k, \boldsymbol{y}_k \rightarrow$ compute $\ell_k(\mathbf{B}_k)$

Hessian Approximation Update via Online Learning

- We aim to minimize $\sum_{k=0}^{N-1} \ell_k(\mathbf{B}_k)$, but we observe ℓ_k after selecting $\mathbf{B}_k$!

  Why? given $\boldsymbol{x}_k, \mathbf{B}_k \to$ select $(\eta_k, \hat{\boldsymbol{x}}_k)$ by BLS $\to$ compute $\boldsymbol{s}_k, \boldsymbol{y}_k \to$ compute $\ell_k(\mathbf{B}_k)$

- Key idea: View the update of $\mathbf{B}_k$ as an online convex opt problem
  - Choose $\mathbf{B}_k \in \mathcal{Z}$, where $\mathcal{Z} = \{\mathbf{B} : \mu \mathbf{I} \preceq \mathbf{B} \preceq L_1 \mathbf{I}\}$
  - Receive $\ell_k(\mathbf{B}_k)$
  - Update $\mathbf{B}_{k+1}$ by an online learning algorithm, e.g., Online Gradient Descent

$$\mathbf{B}_{k+1} = \Pi_{\mathcal{Z}}\left(\mathbf{B}_k - \rho \nabla \ell_k(\mathbf{B}_k)\right)$$

Convergence Analysis

- Recall that

$$\frac{\|\boldsymbol{x}_N - \boldsymbol{x}^*\|^2}{\|\boldsymbol{x}_0 - \boldsymbol{x}^*\|^2} \leq \left(1 + 2\mu\sqrt{\frac{N}{\mathcal{O}(\sum_{k=0}^{N-1} \ell_k(\mathbf{B}_k))}}\right)^{-N}$$

- $\mu \mathbf{I} \preceq \mathbf{B}_k \preceq L_1 \mathbf{I} \;\Rightarrow\; \sum_{k=0}^{N-1} \ell_k(\mathbf{B}_k) \leq L_1^2 N$

Convergence Analysis

- Recall that

$$\frac{\|\boldsymbol{x}_N - \boldsymbol{x}^*\|^2}{\|\boldsymbol{x}_0 - \boldsymbol{x}^*\|^2} \leq \left(1 + 2\mu\sqrt{\frac{N}{\mathcal{O}(\sum_{k=0}^{N-1} \ell_k(\mathbf{B}_k))}}\right)^{-N}$$

- $\mu\mathbf{I} \preceq \mathbf{B}_k \preceq L_1\mathbf{I} \;\Rightarrow\; \sum_{k=0}^{N-1} \ell_k(\mathbf{B}_k) \leq L_1^2 N \;\Rightarrow$ Linear rate

Convergence Analysis

- Recall that

$$\frac{\|\boldsymbol{x}_N - \boldsymbol{x}^*\|^2}{\|\boldsymbol{x}_0 - \boldsymbol{x}^*\|^2} \leq \left(1 + 2\mu\sqrt{\frac{N}{\mathcal{O}(\sum_{k=0}^{N-1} \ell_k(\mathbf{B}_k))}}\right)^{-N}$$

- $\mu\mathbf{I} \preceq \mathbf{B}_k \preceq L_1\mathbf{I} \;\Rightarrow\; \sum_{k=0}^{N-1} \ell_k(\mathbf{B}_k) \leq L_1^2 N \;\Rightarrow$ Linear rate
- A "small-loss" bound by using the smoothness of ℓ_k:

$$\sum_{k=0}^{N-1} \ell_k(\mathbf{B}_k) \leq 4\|\mathbf{B}_0 - \mathbf{H}\|_F^2 + 2\sum_{k=0}^{N-1} \ell_k(\mathbf{H})$$

- A natural choice: $\mathbf{H} = \nabla^2 f(\boldsymbol{x}^*) \Rightarrow \sum_{k=0}^{N-1} \ell_k(\nabla^2 f(\boldsymbol{x}^*)) = \mathcal{O}\left(\frac{L_1 L_2^2 \|\boldsymbol{x}_0 - \boldsymbol{x}^*\|^2}{\mu}\right)$

$$\sum_{k=0}^{N-1} \ell_k(\mathbf{B}_k) = \mathcal{O}\left(\|\mathbf{B}_0 - \nabla^2 f(\boldsymbol{x}^*)\|_F^2 + \frac{L_1 L_2^2 \|\boldsymbol{x}_0 - \boldsymbol{x}^*\|^2}{\mu}\right) \approx L_1^2 d$$

Convergence Analysis

- Recall that

$$\frac{\|\boldsymbol{x}_N - \boldsymbol{x}^*\|^2}{\|\boldsymbol{x}_0 - \boldsymbol{x}^*\|^2} \leq \left(1 + 2\mu\sqrt{\frac{N}{\mathcal{O}(\sum_{k=0}^{N-1} \ell_k(\mathbf{B}_k))}}\right)^{-N}$$

- $\mu \mathbf{I} \preceq \mathbf{B}_k \preceq L_1 \mathbf{I} \;\Rightarrow\; \sum_{k=0}^{N-1} \ell_k(\mathbf{B}_k) \leq L_1^2 N \;\Rightarrow$ Linear rate
- A "small-loss" bound by using the smoothness of ℓ_k:

$$\sum_{k=0}^{N-1} \ell_k(\mathbf{B}_k) \leq 4\|\mathbf{B}_0 - \mathbf{H}\|_F^2 + 2\sum_{k=0}^{N-1} \ell_k(\mathbf{H})$$

- A natural choice: $\mathbf{H} = \nabla^2 f(\boldsymbol{x}^*) \Rightarrow \sum_{k=0}^{N-1} \ell_k(\nabla^2 f(\boldsymbol{x}^*)) = \mathcal{O}\left(\frac{L_1 L_2^2 \|\boldsymbol{x}_0 - \boldsymbol{x}^*\|^2}{\mu}\right)$

$$\sum_{k=0}^{N-1} \ell_k(\mathbf{B}_k) = \mathcal{O}\left(\|\mathbf{B}_0 - \nabla^2 f(\boldsymbol{x}^*)\|_F^2 + \frac{L_1 L_2^2 \|\boldsymbol{x}_0 - \boldsymbol{x}^*\|^2}{\mu}\right) \approx L_1^2 d \;\Rightarrow\; \text{Superlinear rate}$$

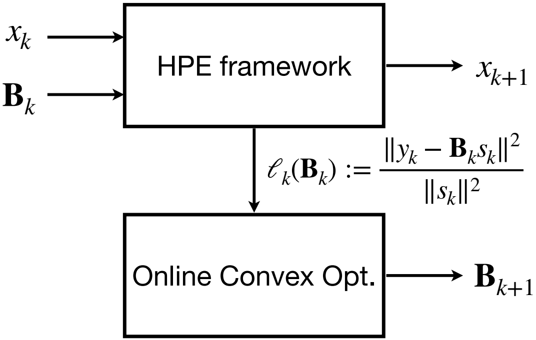
Projection-Free Online Learning

- One issue: Euclidean projection onto $\mathcal{Z} = \{\mathbf{B} : \mu\mathbf{I} \preceq \mathbf{B} \preceq L_1\mathbf{I}\}$ is expensive
  - It requires full eigen-decomposition, which costs $\mathcal{O}(d^3)$

Projection-Free Online Learning

- One issue: Euclidean projection onto $\mathcal{Z} = \{\mathbf{B} : \mu\mathbf{I} \preceq \mathbf{B} \preceq L_1\mathbf{I}\}$ is expensive
  - It requires full eigen-decomposition, which costs $\mathcal{O}(d^3)$
- Solution: We adopted a projection-free approach inspired by [Mhammedi' COLT22]
  - Instead of projection onto the set we only require a separation oracle for the set.

Summary (Strongly Convex)



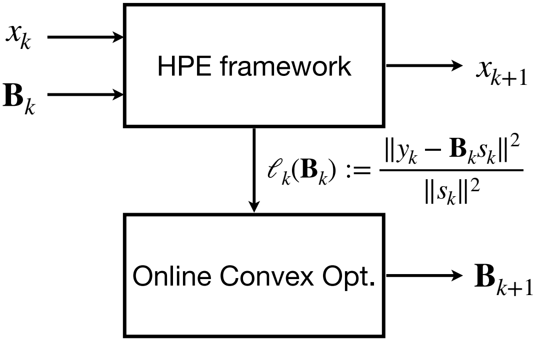
- HPE analysis:

$$\frac{\|\boldsymbol{x}_N - \boldsymbol{x}^*\|^2}{\|\boldsymbol{x}_0 - \boldsymbol{x}^*\|^2} \leq \left(1 + \mu\sqrt{\frac{N}{\sum_{k=0}^{N-1} \ell_k(\mathbf{B}_k)}}\right)^{-N}$$

- Regret analysis:

$$\sum_{k=0}^{N-1} \ell_k(\mathbf{B}_k) \leq \begin{cases} L_1^2 N, \\ \|\mathbf{B}_0 - \mathbf{H}^*\|_F^2 + \sum_{k=0}^{N-1} \ell_k(\mathbf{H}^*) \end{cases}$$

Summary (Strongly Convex)



- HPE analysis:

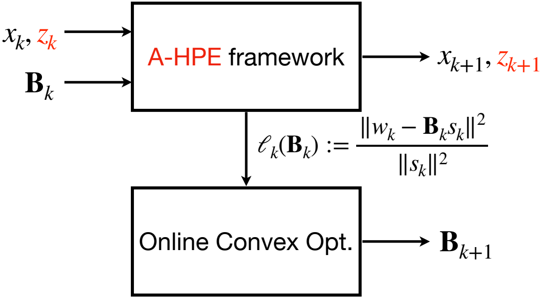
$$\frac{\|\boldsymbol{x}_N - \boldsymbol{x}^*\|^2}{\|\boldsymbol{x}_0 - \boldsymbol{x}^*\|^2} \leq \left(1 + \mu\sqrt{\frac{N}{\sum_{k=0}^{N-1} \ell_k(\mathbf{B}_k)}}\right)^{-N}$$

- Regret analysis:

$$\sum_{k=0}^{N-1} \ell_k(\mathbf{B}_k) \leq \begin{cases} L_1^2 N, \\ \|\mathbf{B}_0 - \mathbf{H}^*\|_F^2 + \sum_{k=0}^{N-1} \ell_k(\mathbf{H}^*) \end{cases}$$

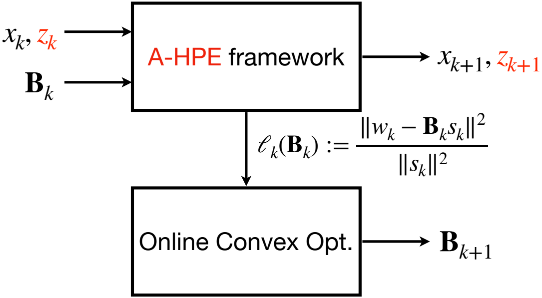
- Omitted details:
  - Backtracking line search
  - Approx. linear solver
  - Projecton-free online learning

What about the CVX setting?



- In the CVX setting, we can use the accelerated HPE [Monteiro-Svaiter'13]

What about the CVX setting?

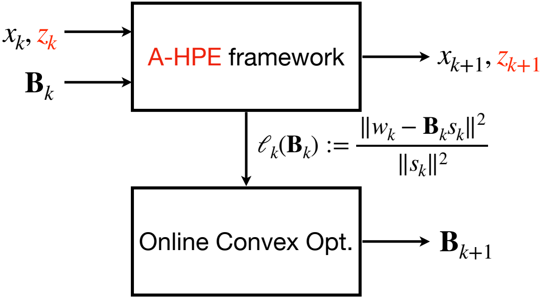


- In the CVX setting, we can use the accelerated HPE [Monteiro-Svaiter'13]
- A-HPE analysis:

$$f(\mathbf{x}_N) - f(\mathbf{x}^*) \leq \frac{2\|\mathbf{z}_0 - \mathbf{x}^*\|^2}{N^{2.5}} \sqrt{\sum_{k=0}^{N-1} \ell_k(\mathbf{B}_k)},$$

$$\mathbf{w}_k = \nabla f(\mathbf{x}_{k+1}) - \nabla f(\mathbf{y}_k), \ \mathbf{s}_k = \mathbf{x}_{k+1} - \mathbf{y}_k$$

What about the CVX setting?



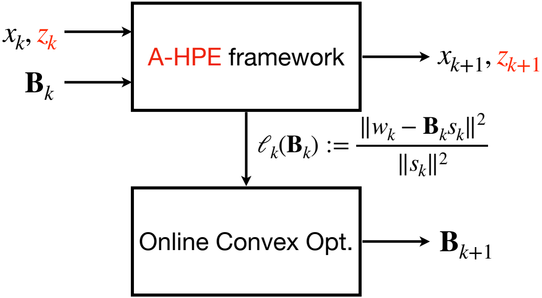
- In the CVX setting, we can use the accelerated HPE [Monteiro-Svaiter'13]
- A-HPE analysis:

$$f(\mathbf{x}_N) - f(\mathbf{x}^*) \leq \frac{2\|\mathbf{z}_0 - \mathbf{x}^*\|^2}{N^{2.5}} \sqrt{\sum_{k=0}^{N-1} \ell_k(\mathbf{B}_k)},$$

$$\mathbf{w}_k = \nabla f(\mathbf{x}_{k+1}) - \nabla f(\mathbf{y}_k), \; \mathbf{s}_k = \mathbf{x}_{k+1} - \mathbf{y}_k$$

- Regret analysis:

$$\sum_{k=0}^{N-1} \ell_k(\mathbf{B}_k) \leq \begin{cases} L_1^2 N, \\ ??? \end{cases}$$



- In the CVX setting, we can use the accelerated HPE [Monteiro-Svaiter'13]
- A-HPE analysis:

$$f(\mathbf{x}_N) - f(\mathbf{x}^*) \leq \frac{2\|\mathbf{z}_0 - \mathbf{x}^*\|^2}{N^{2.5}} \sqrt{\sum_{k=0}^{N-1} \ell_k(\mathbf{B}_k)},$$

$$\mathbf{w}_k = \nabla f(\mathbf{x}_{k+1}) - \nabla f(\mathbf{y}_k), \ \mathbf{s}_k = \mathbf{x}_{k+1} - \mathbf{y}_k$$

- Regret analysis:

$$\sum_{k=0}^{N-1} \ell_k(\mathbf{B}_k) \leq \begin{cases} L_1^2 N, \\ ??? \end{cases}$$

Static regret $\Rightarrow$ dynamic regret

$\tilde{\mathcal{O}}(\sqrt{d}/k^{2.5})$ Convergence Rate

- Recall that in the strongly convex setting, we use the regret bound

$$\sum_{k=0}^{N-1} \ell_k(\mathbf{B}_k) \leq 4\|\mathbf{B}_0 - \mathbf{H}^*\|_F^2 + 2\sum_{k=0}^{N-1} \ell_k(\mathbf{H}^*),$$

  where $\mathbf{H}^* = \nabla^2 f(\boldsymbol{x}^*)$
- By linear convergence, we have

$$\sum_{k=0}^{N-1} \ell_k(\mathbf{H}^*) \leq \sum_{k=0}^{N-1} L_2^2\|\boldsymbol{x}_k - \boldsymbol{x}^*\|^2 = \mathcal{O}\left(\frac{L_1 L_2^2}{\mu}\|\boldsymbol{x}_0 - \boldsymbol{x}^*\|^2\right)$$

- However, we do not have linear convergence in the convex setting!
- Have to take a different approach via dynamic regret bound

$\tilde{\mathcal{O}}(\sqrt{d}/k^{2.5})$ Convergence Rate

- For any sequence $\{\mathbf{H}_k\}_{k=0}^{N-1}$ with $\mathbf{H}_k \in \mathcal{Z} = \{\mathbf{B} : 0 \preceq \mathbf{B} \preceq L_1 \mathbf{I}\}$, we can show that

$$\sum_{k=0}^{N-1} \ell_k(\mathbf{B}_k) = \mathcal{O}\left(\|\mathbf{B}_0 - \mathbf{H}_0\|_F^2 + \sum_{k=0}^{N-1} \ell_k(\mathbf{H}_k) + L_1 \sqrt{d} \sum_{k=0}^{N-1} \|\mathbf{H}_{k+1} - \mathbf{H}_k\|_F \right)$$

- We then choose $\mathbf{H}_k = \nabla^2 f(\boldsymbol{y}_k)$ for $k = 0, \ldots, N-1$
- With careful potential analysis, we can bound

$$\sum_{k=0}^{N-1} \ell_k(\mathbf{H}_k) = \mathcal{O}\left(L_2^2 \|\boldsymbol{z}_0 - \boldsymbol{x}^*\|^2 \right), \quad \sum_{k=0}^{N-1} \|\mathbf{H}_{k+1} - \mathbf{H}_k\|_F = \mathcal{O}\left(L_2 \sqrt{d} \|\boldsymbol{z}_0 - \boldsymbol{x}^*\| \log N \right)$$

$\tilde{\mathcal{O}}(\sqrt{d}/k^{2.5})$ Convergence Rate

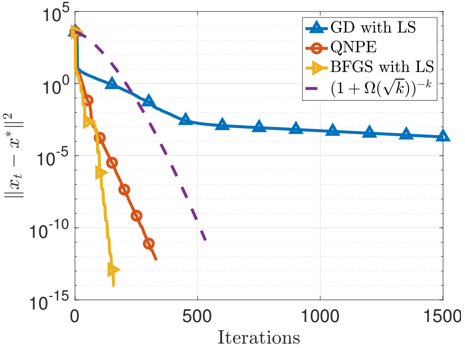
- Putting everything together:

$$\sum_{k=0}^{N-1} \ell_k(\mathbf{B}_k) = \mathcal{O}\left(\|\mathbf{B}_0 - \mathbf{H}_0\|_F^2 + \sum_{k=0}^{N-1} \ell_k(\mathbf{H}_k) + L_1\sqrt{d}\sum_{k=0}^{N-1} \|\mathbf{H}_{k+1} - \mathbf{H}_k\|_F \right)$$
$$= \mathcal{O}\left(\|\mathbf{B}_0 - \nabla^2 f(\boldsymbol{x}_0)\|_F^2 + L_2^2\|\boldsymbol{x}_0 - \boldsymbol{x}^*\|^2 + L_1 L_2 d \|\boldsymbol{z}_0 - \boldsymbol{x}^*\| \log N \right)$$

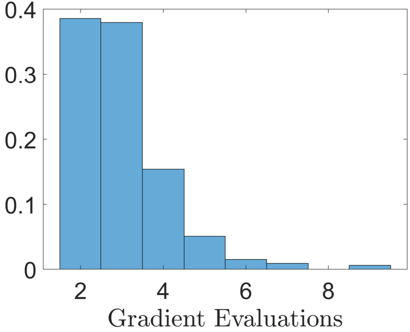
- Thus, we have

$$f(\boldsymbol{x}_N) - f(\boldsymbol{x}^*) \leq \frac{2\|\boldsymbol{z}_0 - \boldsymbol{x}^*\|^2}{N^{2.5}} \sqrt{\mathcal{O}\left(\sum_{k=0}^{N-1} \ell_k(\mathbf{B}_k) \right)} = \mathcal{O}\left(\frac{\sqrt{d}\log N}{N^{2.5}} \right)$$

Numerical Experiment (Strongly Convex Setting)



(a) Convergence by iteration

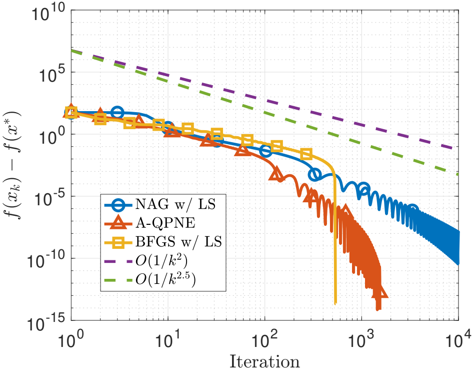


(b) Histogram of gradient evals

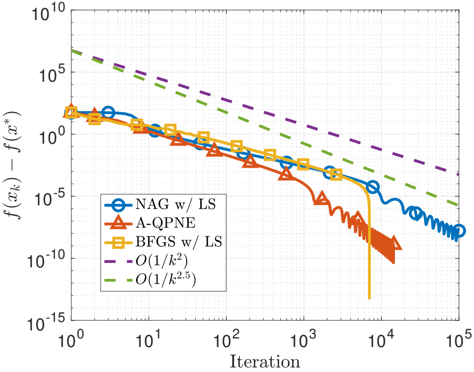
Figure: Numerical results for an L_2-regularized logistic regression problem

Numerical Experiment (Convex Setting)

Numerical results for the hard a convex cubic-quadratic problem



(a) Dimension $d = 50$



(b) Dimension $d = 500$

How about the Nonconvex Setting?

Goal: find $\boldsymbol{x}$ such that $\|\nabla f(\boldsymbol{x})\|_2 \leq \epsilon$

How about the Nonconvex Setting?

Goal: find $\boldsymbol{x}$ such that $\|\nabla f(\boldsymbol{x})\|_2 \leq \epsilon$

- If ∇f is Lipschitz, GD has a complexity of $\mathcal{O}(\epsilon^{-2})$, which is optimal!

How about the Nonconvex Setting?

Goal: find $\boldsymbol{x}$ such that $\|\nabla f(\boldsymbol{x})\|_2 \leq \epsilon$

- If ∇f is Lipschitz, GD has a complexity of $\mathcal{O}(\epsilon^{-2})$, which is optimal!
- What if ∇f and $\nabla^2 f$ are both Lipschitz? (but we only have access to ∇f)
  - Two concurrent works achieved a grad complexity of $O(\epsilon^{-7/4}\log(1/\epsilon))$ [Carmon,Duchi,Hinder,Sidford, ICML'17] & [Agarwal,AllenZhu,Bullins,Hazan,Ma, STOC'17].
  - Later, [Li,Lin, ICML'22] shaved the log term and obtained $O(\epsilon^{-7/4})$

How about the Nonconvex Setting?

Goal: find $\boldsymbol{x}$ such that $\|\nabla f(\boldsymbol{x})\|_2 \leq \epsilon$

- If ∇f is Lipschitz, GD has a complexity of $\mathcal{O}(\epsilon^{-2})$, which is optimal!

- What if ∇f and $\nabla^2 f$ are both Lipschitz? (but we only have access to ∇f)
  - Two concurrent works achieved a grad complexity of $O(\epsilon^{-7/4}\log(1/\epsilon))$ [Carmon,Duchi,Hinder,Sidford, ICML'17] & [Agarwal,AllenZhu,Bullins,Hazan,Ma, STOC'17].
  - Later, [Li,Lin, ICML'22] shaved the log term and obtained $O(\epsilon^{-7/4})$

- Our Result: gradient complexity of $O(d^{1/4}\epsilon^{-13/8})$, where d is the problem dimension.

How about the Nonconvex Setting?

Goal: find $\boldsymbol{x}$ such that $\|\nabla f(\boldsymbol{x})\|_2 \leq \epsilon$

- If ∇f is Lipschitz, GD has a complexity of $\mathcal{O}(\epsilon^{-2})$, which is optimal!
- What if ∇f and $\nabla^2 f$ are both Lipschitz? (but we only have access to ∇f)
  - Two concurrent works achieved a grad complexity of $O(\epsilon^{-7/4}\log(1/\epsilon))$ [Carmon,Duchi,Hinder,Sidford, ICML'17] & [Agarwal,AllenZhu,Bullins,Hazan,Ma, STOC'17].
  - Later, [Li,Lin, ICML'22] shaved the log term and obtained $O(\epsilon^{-7/4})$
- Our Result: gradient complexity of $O(d^{1/4}\epsilon^{-13/8})$, where d is the problem dimension.
- For $d \leq \frac{1}{\sqrt{\epsilon}}$, our iteration complexity outperforms existing first-order methods.

Main Ideas

- We design a QN-type algorithm that incorporates solving **two online learning problems** under the hood!

Main Ideas

- We design a QN-type algorithm that incorporates solving **two online learning problems** under the hood!
- We first use the **Online-to-Nonconvex framework** of [Cutkosky,Mehta,Orabona, ICML'23] to reformulate the task of finding a stationary point of a nonconvex function as an OCO

Main Ideas

- We design a QN-type algorithm that incorporates solving **two online learning problems** under the hood!
- We first use the **Online-to-Nonconvex framework** of [Cutkosky,Mehta,Orabona, ICML'23] to reformulate the task of finding a stationary point of a nonconvex function as an OCO
- Then, we introduce a novel **Optimistic Quasi-Newton** method for solving the OCO
  - The Hessian approximation update itself is framed as an online learning problem in the space of matrices. (similar to the previous settings!)

References

- R. Jiang, Q. Jin, A. Mokhtari, "Online Learning Guided Curvature Approximation: A Quasi-Newton Method with Global Non-Asymptotic Superlinear Convergence," COLT 2023.
- R. Jiang, A. Mokhtari, "Accelerated Quasi-Newton Proximal Extragradient: Faster Rate for Smooth Convex Optimization," NeurIPS 2023 (Spotlight).
- R. Jiang*, A. Mokhtari*, F. Patitucci* "Improved Complexity for Smooth Nonconvex Optimization: A Two-Level Online Learning Approach with Quasi-Newton Methods," Arxiv, Dec. 2024.

Thank you!