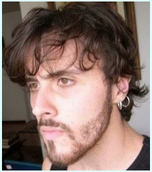
Level Set Teleportation: an Optimization Perspective

Aaron Mishkin   Alberto Bietti   Robert M. Gower

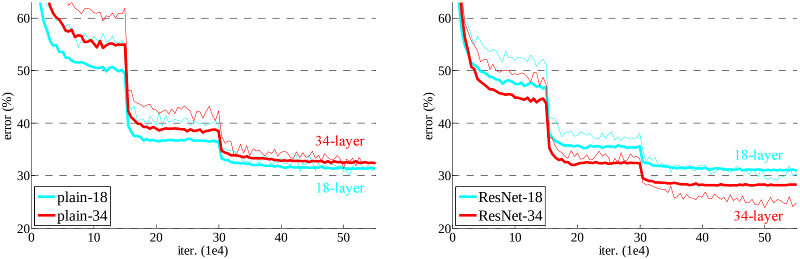
Introduction: Plateau's and Sudden Drops

Basic Problem: deep-learning objectives have many flat regions where the gradient is small and optimization is slow [FA00].

https://arxiv.org/abs/1512.03385

Introduction: Plateau's and Sudden Drops

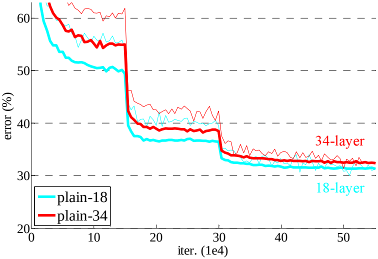
Basic Problem: deep-learning objectives have many flat regions where the gradient is small and optimization is slow [FA00].

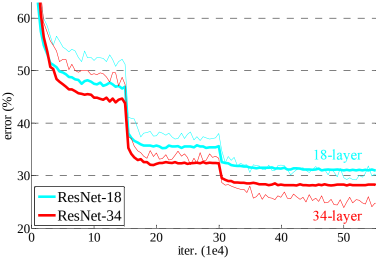


https://arxiv.org/abs/1512.03385

Introduction: Plateau's and Sudden Drops

Basic Problem: deep-learning objectives have many flat regions where the gradient is small and optimization is slow [FA00].





Flat regions can cause plateaus in training loss and then sudden drops when iterates finally escape.

https://arxiv.org/abs/1512.03385

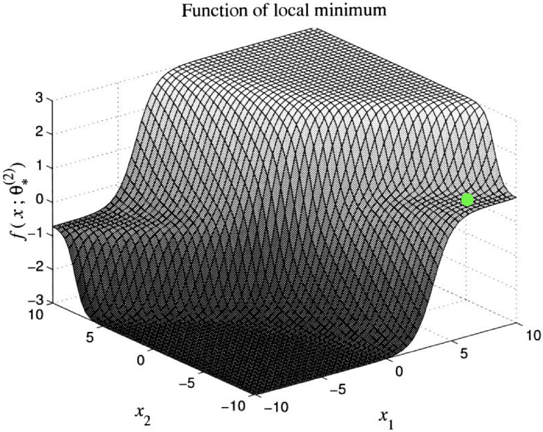
Introduction: Flat Loss Surfaces

Faster optimization requires escaping flat regions quickly.

https://www.sciencedirect.com/science/article/abs/pii/S0893608000000095

Introduction: Flat Loss Surfaces

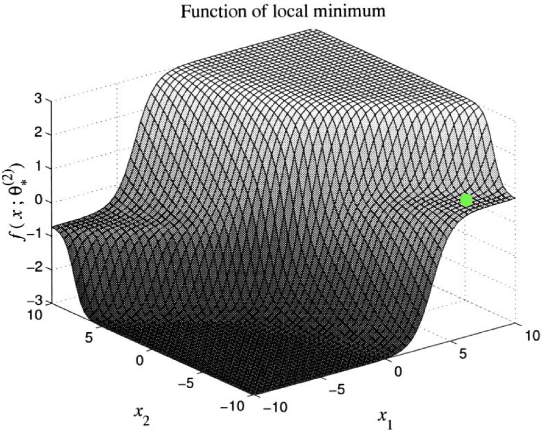
Faster optimization requires escaping flat regions quickly.



https://www.sciencedirect.com/science/article/abs/pii/S0893608000000095

Introduction: Flat Loss Surfaces

Faster optimization requires escaping flat regions quickly.



- Newton's method could do this, but Newton doesn't work for non-convex objectives due to negative curvature.

https://www.sciencedirect.com/science/article/abs/pii/S0893608000000095

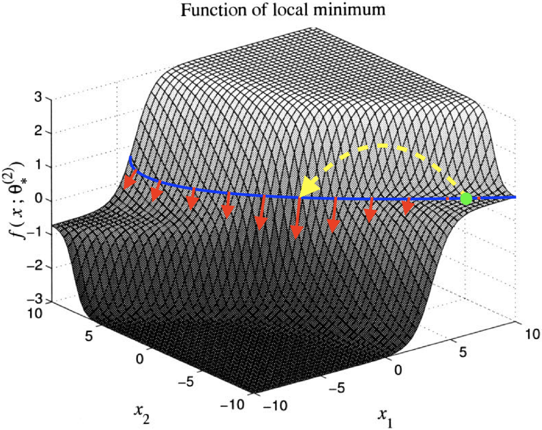
Introduction: Towards Teleportation

What we really want to do is jump away to a big gradient!

https://www.sciencedirect.com/science/article/abs/pii/S0893608000000095

Introduction: Towards Teleportation

What we really want to do is jump away to a big gradient!

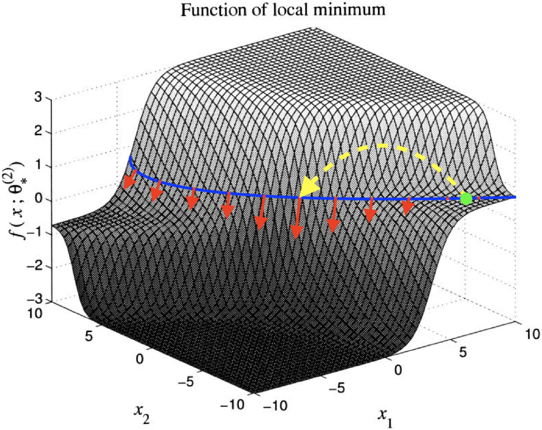


---

https://www.sciencedirect.com/science/article/abs/pii/S0893608000000095

Introduction: Towards Teleportation

What we really want to do is jump away to a big gradient!



This is the picture behind **level set teleportation** [Zha+23b].

https://www.sciencedirect.com/science/article/abs/pii/S0893608000000095

Introduction: Background

Teleport incipit: Armenta et al. [Arm+20] and Armenta and Jodoin [AJ21] propose random jumps using symmetry operators.

Introduction: Background

Teleport incipit: Armenta et al. [Arm+20] and Armenta and Jodoin [AJ21] propose random jumps using symmetry operators.

- But it doesn't work very well...

Introduction: Background

Teleport incipit: Armenta et al. [Arm+20] and Armenta and Jodoin [AJ21] propose random jumps using symmetry operators.

- But it doesn't work very well. . .

Enter optimization: Zhao et al. [Zha+23a] optimize over symmetries and alternate between GD and teleportation steps.

Introduction: Background

Teleport incipit: Armenta et al. [Arm+20] and Armenta and Jodoin [AJ21] propose random jumps using symmetry operators.

- But it doesn't work very well...

Enter optimization: Zhao et al. [Zha+23a] optimize over symmetries and alternate between GD and teleportation steps.

- They use Newton's method to prove a mixed linear/quadratic rate for strongly-convex functions.

Introduction: Background

Teleport incipit: Armenta et al. [Arm+20] and Armenta and Jodoin [AJ21] propose random jumps using symmetry operators.

- But it doesn't work very well...

Enter optimization: Zhao et al. [Zha+23a] optimize over symmetries and alternate between GD and teleportation steps.

- They use Newton's method to prove a mixed linear/quadratic rate for strongly-convex functions.
- They give standard rates under the PL-condition [KNS16] and slightly stronger guarantees for non-convex functions.

Introduction: Background

Teleport incipit: Armenta et al. [Arm+20] and Armenta and Jodoin [AJ21] propose random jumps using symmetry operators.

- But it doesn't work very well...

Enter optimization: Zhao et al. [Zha+23a] optimize over symmetries and alternate between GD and teleportation steps.

- They use Newton's method to prove a mixed linear/quadratic rate for strongly-convex functions.
- They give standard rates under the PL-condition [KNS16] and slightly stronger guarantees for non-convex functions.
- But, symmetries only approximate teleportation...

Introduction: Background

Teleport incipit: Armenta et al. [Arm+20] and Armenta and Jodoin [AJ21] propose random jumps using symmetry operators.

- But it doesn't work very well...

Enter optimization: Zhao et al. [Zha+23a] optimize over symmetries and alternate between GD and teleportation steps.

- They use Newton's method to prove a mixed linear/quadratic rate for strongly-convex functions.
- They give standard rates under the PL-condition [KNS16] and slightly stronger guarantees for non-convex functions.
- But, symmetries only approximate teleportation...
- And nothing is known for non-strongly convex functions.

Introduction: Background

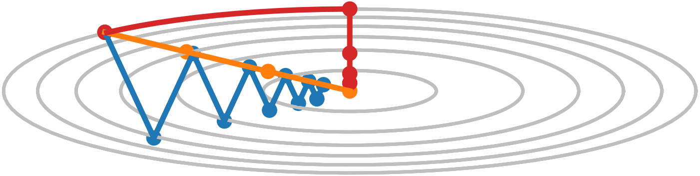
Teleport incipit: Armenta et al. [Arm+20] and Armenta and Jodoin [AJ21] propose random jumps using symmetry operators.

- But it doesn't work very well...

Enter optimization: Zhao et al. [Zha+23a] optimize over symmetries and alternate between GD and teleportation steps.

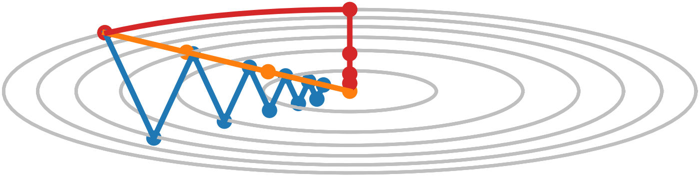
- They use Newton's method to prove a mixed linear/quadratic rate for strongly-convex functions.
- They give standard rates under the PL-condition [KNS16] and slightly stronger guarantees for non-convex functions.
- But, symmetries only approximate teleportation...
- And nothing is known for non-strongly convex functions.

Introduction: Contributions



Our Contributions:

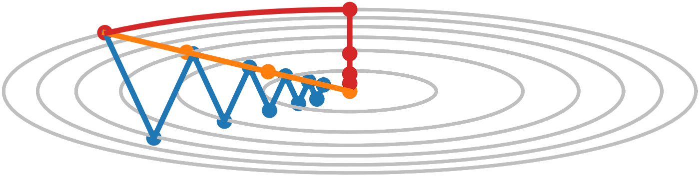
Introduction: Contributions



Our Contributions:

- We show teleportation only accelerates optimization when (i) there is curvature and (ii) adaptive step-sizes are used.

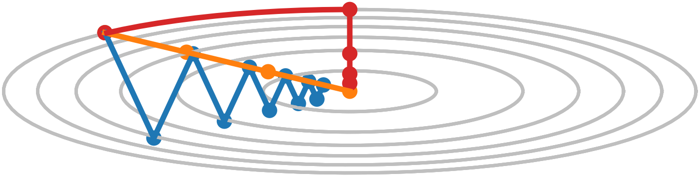
Introduction: Contributions



Our Contributions:

- We show teleportation only accelerates optimization when (i) there is curvature and (ii) adaptive step-sizes are used.
- We show teleportation speeds-up optimization under Hessian stability (rates faster than $O(1/K)$!).

Introduction: Contributions



Our Contributions:

- We show teleportation only accelerates optimization when (i) there is curvature and (ii) adaptive step-sizes are used.
- We show teleportation speeds-up optimization under Hessian stability (rates faster than $O(1/K)$!).
- We develop a fast, parameter-free algorithm for solving teleportation problems.

Optimization Background

Optimization Setting

Goal: minimize an objective $f : \mathbb{R}^d \to \mathbb{R}$,

$$\mathcal{W}^* = \arg\min_w f(w).$$

Optimization Setting

Goal: minimize an objective $f : \mathbb{R}^d \to \mathbb{R}$,

$$\mathcal{W}^* = \arg\min_{w} f(w).$$

We are willing to make the following assumptions.

Optimization Setting

Goal: minimize an objective $f : \mathbb{R}^d \to \mathbb{R}$,

$$\mathcal{W}^* = \arg\min_{w} f(w).$$

We are willing to make the following assumptions.

- f is convex, meaning for every $y, x \in \mathbb{R}^d$,

$$f(y) \geq f(x) + \langle \nabla f(x), y - x \rangle .$$

Optimization Setting

Goal: minimize an objective $f : \mathbb{R}^d \to \mathbb{R}$,

$$\mathcal{W}^* = \arg\min_w f(w).$$

We are willing to make the following assumptions.

- f is convex, meaning for every $y, x \in \mathbb{R}^d$,

$$f(y) \geq f(x) + \langle \nabla f(x), y - x \rangle .$$

- f is differentiable and ∇f is L-Lipschitz: $\forall x, y \in \mathbb{R}^d$,

$$\|\nabla f(y) - \nabla f(x)\|_2 \leq L\|y - x\|_2,$$

Optimization Setting

Goal: minimize an objective $f : \mathbb{R}^d \to \mathbb{R}$,

$$\mathcal{W}^* = \arg\min_w f(w).$$

We are willing to make the following assumptions.

- f is convex, meaning for every $y, x \in \mathbb{R}^d$,

$$f(y) \geq f(x) + \langle \nabla f(x), y - x \rangle .$$

- f is differentiable and ∇f is L-Lipschitz: $\forall x, y \in \mathbb{R}^d$,

$$\|\nabla f(y) - \nabla f(x)\|_2 \leq L\|y - x\|_2,$$

  - ∇f doesn't change too fast...

Lipschitz Gradients: Motivation

Why do we need ∇f to be L-Lipschitz?

Lipschitz Gradients: Motivation

Why do we need ∇f to be L-Lipschitz?

Intuition: gradient magnitude isn't informative for non-smooth f.

$$\textbf{GD}: \; w_{k+1} = w_k - \eta \nabla f(w_k).$$

Lipschitz Gradients: Motivation

Why do we need ∇f to be L-Lipschitz?

Intuition: gradient magnitude isn't informative for non-smooth f.

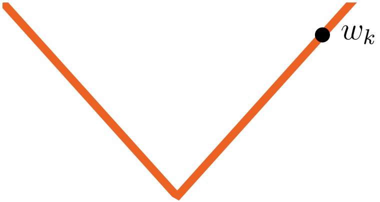
$$\textbf{GD}: \ w_{k+1} = w_k - \eta \nabla f(w_k).$$

Non-Smooth

Why do we need ∇f to be L-Lipschitz?

Intuition: gradient magnitude isn't informative for non-smooth f.

$$\textbf{GD}: \; w_{k+1} = w_k - \eta \nabla f(w_k).$$



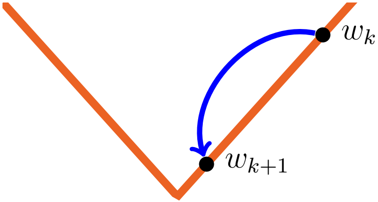
Non-Smooth

Lipschitz Gradients: Motivation

Why do we need ∇f to be L-Lipschitz?

Intuition: gradient magnitude isn't informative for non-smooth f.

$$\textbf{GD}: \; w_{k+1} = w_k - \eta \nabla f(w_k).$$



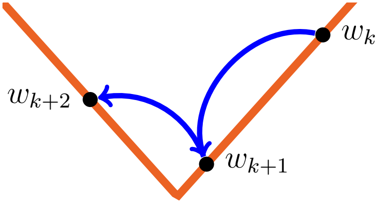
Non-Smooth

Lipschitz Gradients: Motivation

Why do we need ∇f to be L-Lipschitz?

Intuition: gradient magnitude isn't informative for non-smooth f.

$$\textbf{GD}: \; w_{k+1} = w_k - \eta \nabla f(w_k).$$



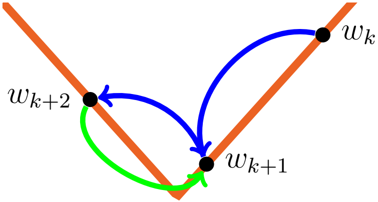
Non-Smooth

Lipschitz Gradients: Motivation

Why do we need ∇f to be L-Lipschitz?

Intuition: gradient magnitude isn't informative for non-smooth f.

$$\textbf{GD}: \; w_{k+1} = w_k - \eta \nabla f(w_k).$$

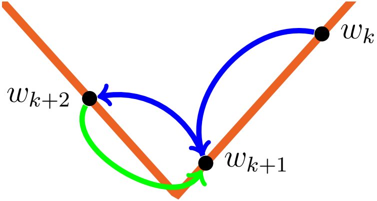


Non-Smooth

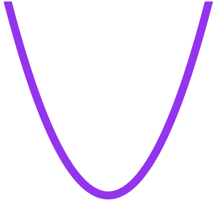
Why do we need ∇f to be L-Lipschitz?

Intuition: gradient magnitude isn't informative for non-smooth f.

$$\textbf{GD}: \; w_{k+1} = w_k - \eta \nabla f(w_k).$$



Non-Smooth

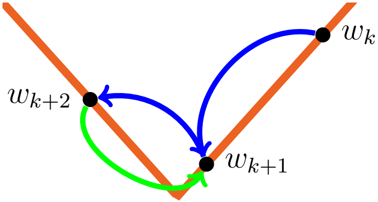


Smooth

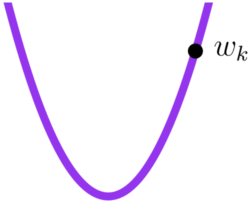
Why do we need ∇f to be L-Lipschitz?

Intuition: gradient magnitude isn't informative for non-smooth f.

$$\textbf{GD}: \; w_{k+1} = w_k - \eta \nabla f(w_k).$$



Non-Smooth



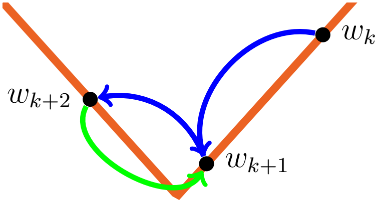
Smooth

Lipschitz Gradients: Motivation

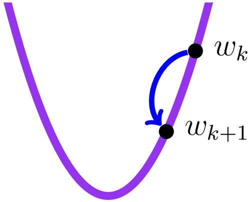
Why do we need ∇f to be L-Lipschitz?

Intuition: gradient magnitude isn't informative for non-smooth f.

$$\textbf{GD}: \ w_{k+1} = w_k - \eta \nabla f(w_k).$$



Non-Smooth



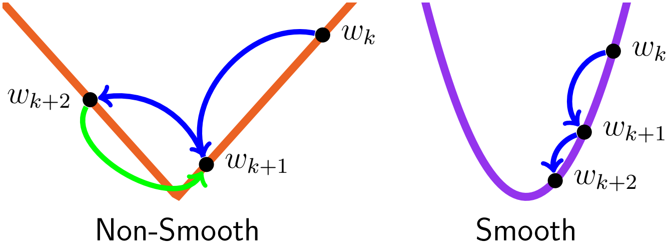
Smooth

Lipschitz Gradients: Motivation

Why do we need ∇f to be L-Lipschitz?

Intuition: gradient magnitude isn't informative for non-smooth f.

$$\textbf{GD}: \; w_{k+1} = w_k - \eta \nabla f(w_k).$$



Lipschitz Gradients: Upper Bounds

If ∇f is L-Lipschitz, then

$$|f(y) - f(x) - \langle \nabla f(x), y - x \rangle|$$

Lipschitz Gradients: Upper Bounds

If ∇f is L-Lipschitz, then

$$\begin{aligned}
&|f(y) - f(x) - \langle \nabla f(x), y - x \rangle| \\
&\qquad = \left| \int_0^1 \langle \nabla f(x + t(y - x)) - \nabla f(x), y - x \rangle \, dt \right|
\end{aligned}$$

Lipschitz Gradients: Upper Bounds

If ∇f is L-Lipschitz, then

$$\begin{aligned}
&|f(y) - f(x) - \langle \nabla f(x), y - x\rangle| \\
&\qquad = \left| \int_0^1 \langle \nabla f(x + t(y-x)) - \nabla f(x), y - x\rangle \, dt \right| \\
&\qquad \leq \int_0^1 \|\nabla f(x + t(y-x)) - \nabla f(x)\|_2 \|y - x\|_2 dt
\end{aligned}$$

Lipschitz Gradients: Upper Bounds

If ∇f is L-Lipschitz, then

$$
\begin{aligned}
&|f(y) - f(x) - \langle \nabla f(x), y - x \rangle| \\
&\qquad = \left| \int_0^1 \langle \nabla f(x + t(y - x)) - \nabla f(x), y - x \rangle \, dt \right| \\
&\qquad \leq \int_0^1 \|\nabla f(x + t(y - x)) - \nabla f(x)\|_2 \|y - x\|_2 dt \\
&\qquad \leq \int_0^1 Lt \|y - x\|_2 \|y - x\|_2 dt
\end{aligned}
$$

Lipschitz Gradients: Upper Bounds

If ∇f is L-Lipschitz, then

$$\begin{aligned}
&|f(y) - f(x) - \langle \nabla f(x), y - x \rangle| \\
&\qquad = \left| \int_0^1 \langle \nabla f(x + t(y - x)) - \nabla f(x), y - x \rangle \, dt \right| \\
&\qquad \leq \int_0^1 \|\nabla f(x + t(y - x)) - \nabla f(x)\|_2 \|y - x\|_2 dt \\
&\qquad \leq \int_0^1 Lt\|y - x\|_2 \|y - x\|_2 dt \\
&\qquad \leq \frac{L}{2} \|y - x\|_2^2.
\end{aligned}$$

Lipschitz Gradients: Upper Bounds

If ∇f is L-Lipschitz, then

$$\begin{aligned} |f(y) - f(x) - \langle \nabla f(x), y - x \rangle| \\ &= \left| \int_0^1 \langle \nabla f(x + t(y-x)) - \nabla f(x), y - x \rangle \, dt \right| \\ &\leq \int_0^1 \|\nabla f(x + t(y-x)) - \nabla f(x)\|_2 \|y - x\|_2 dt \\ &\leq \int_0^1 Lt\|y - x\|_2 \|y - x\|_2 dt \\ &\leq \frac{L}{2}\|y - x\|_2^2. \end{aligned}$$

The objective is upper-bounded by a quadratic function!

$$f(y) \leq f(x) + \langle \nabla f(x), y - x \rangle + \frac{L}{2}\|y - x\|_2^2.$$

Lipschitz Gradients: L-Smoothness

We say f is L-smooth if for every $x, y \in \mathbb{R}^d$,

$$f(y) \leq Q_x(y) := f(x) + \langle \nabla f(x), y - x \rangle + \frac{L}{2}\|y - x\|_2^2.$$

Lipschitz Gradients: L-Smoothness

We say f is L-smooth if for every $x, y \in \mathbb{R}^d$,

$$f(y) \leq Q_x(y) := f(x) + \langle \nabla f(x), y - x \rangle + \frac{L}{2}\|y - x\|_2^2.$$

The function $Q_x(y)$ is a majorant of f. Minimizing $Q_x(y)$ in y gives,

$$\nabla_y Q_x(\hat{y}) = \nabla f(x) + L(\hat{y} - x) = 0$$

Lipschitz Gradients: L-Smoothness

We say f is L-smooth if for every $x, y \in \mathbb{R}^d$,

$$f(y) \leq Q_x(y) := f(x) + \langle \nabla f(x), y - x \rangle + \frac{L}{2}\|y - x\|_2^2.$$

The function $Q_x(y)$ is a majorant of f. Minimizing $Q_x(y)$ in y gives,

$$\begin{aligned} \nabla_y Q_x(\hat{y}) &= \nabla f(x) + L(\hat{y} - x) = 0 \\ \implies \hat{y} &= x - \frac{1}{L}\nabla f(x). \end{aligned}$$

Lipschitz Gradients: L-Smoothness

We say f is L-smooth if for every $x, y \in \mathbb{R}^d$,

$$f(y) \leq Q_x(y) := f(x) + \langle \nabla f(x), y - x \rangle + \frac{L}{2}\|y - x\|_2^2.$$

The function $Q_x(y)$ is a majorant of f. Minimizing $Q_x(y)$ in y gives,

$$\nabla_y Q_x(\hat{y}) = \nabla f(x) + L(\hat{y} - x) = 0$$
$$\implies \hat{y} = x - \frac{1}{L}\nabla f(x).$$

Gradient descent is a majorization-minimization algorithm!

Lipschitz Gradients: Majorization-Minimization

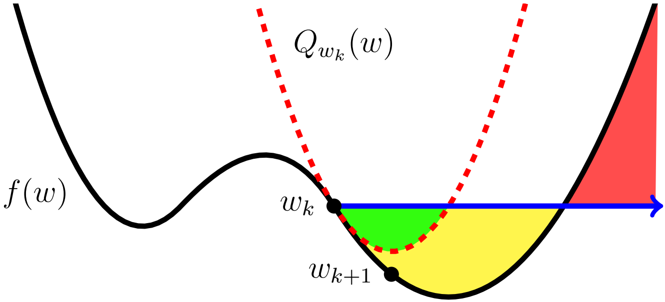
Using $w_{k+1} = w_k - \frac{1}{L}\nabla f(x)$ in Q_{w_k} gives guaranteed progress.

$$\textbf{Descent Lemma}: f(w_{k+1}) \leq f(w_k) - \frac{1}{L}\|\nabla f(w_k)\|_2^2.$$

Lipschitz Gradients: Majorization-Minimization

Using $w_{k+1} = w_k - \frac{1}{L}\nabla f(x)$ in Q_{w_k} gives guaranteed progress.

$$\textbf{Descent Lemma}: f(w_{k+1}) \leq f(w_k) - \frac{1}{L}\|\nabla f(w_k)\|_2^2.$$



Level Set Teleportation

Basic Idea: L-smoothness relates gradient magnitude to descent in function values,

$$f(w_{k+1}) \leq f(w_k) - \frac{1}{L}\|\nabla f(w_k)\|_2^2.$$

- All other quantities held constant, maximizing $\|\nabla f(w_k)\|_2$ maximizes guaranteed progress.

Level Set Teleportation

Basic Idea: L-smoothness relates gradient magnitude to descent in function values,

$$f(w_{k+1}) \leq f(w_k) - \frac{1}{L}\|\nabla f(w_k)\|_2^2.$$

- All other quantities held constant, maximizing $\|\nabla f(w_k)\|_2$ maximizes guaranteed progress.

This lets us formalize our picture version of **level set teleportation**,

$$w_k^+ \in \arg\max_{w} \frac{1}{2}\|\nabla f(w)\|_2^2 \quad \text{s.t.} \quad f(w) \leq f(w_k).$$

Level Set Teleportation.

Level Set Teleportation: Algorithm

- Let $\mathcal{B} \subset \mathbb{N}$ be start indices for teleportation blocks.

Level Set Teleportation: Algorithm

- Let $\mathcal{B} \subset \mathbb{N}$ be start indices for teleportation blocks.
- Each teleportation block $i \in \mathcal{B}$ consists of $b_i \geq 1$ steps.

Level Set Teleportation: Algorithm

- Let $\mathcal{B} \subset \mathbb{N}$ be start indices for teleportation blocks.
- Each teleportation block $i \in \mathcal{B}$ consists of $b_i \geq 1$ steps.
- Complete teleportation schedule is $\mathcal{T}$.

Algorithm GD with Teleportation

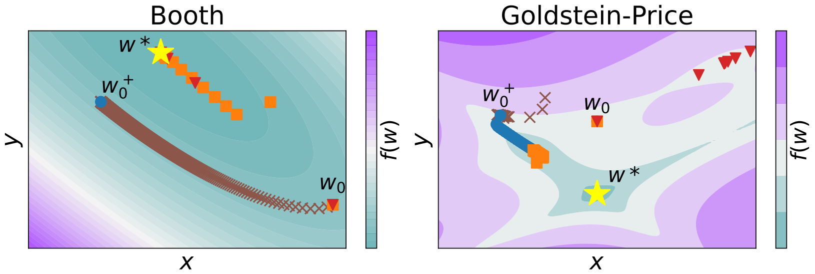
Inputs: w_0; step-sizes η_k; block indices $\mathcal{B}$, sizes b_k.
$\mathcal{T} \leftarrow \bigcup_{k \in \mathcal{B}} \{k, k+1, \ldots, k + b_k - 1\}$
for $k \in \{0, \ldots, K\}$ **do**
  if $k \in \mathcal{T}$ **then**
    $w_k^+ \in \arg\max \{\|\nabla f(w)\|_2 : f(w) \leq f(w_k)\}$
  else
    $w_k^+ \leftarrow w_k$
  end if
  $w_{k+1} \leftarrow w_k^+ - \eta_k \nabla f(w_k^+)$
end for
Output: w_K

Level Set Teleportation: Test Functions

$$w_k^+ \in \arg\max_w \frac{1}{2}\|\nabla f(w)\|_2^2 \quad \text{s.t.} \quad f(w) \leq f(w_k).$$

Level Set Teleportation: Test Functions

$$w_k^+ \in \arg\max_{w} \frac{1}{2}\|\nabla f(w)\|_2^2 \quad \text{s.t.} \quad f(w) \leq f(w_k).$$



Teleportation in action on two test functions.

Level Set Teleportation: Newton's Method

$$w_k^+ \in \arg\max_w \frac{1}{2}\|\nabla f(w)\|_2^2 \quad \text{s.t.} \quad f(w) \le f(w_k).$$

Level Set Teleportation: Newton's Method

$$w_k^+ \in \arg\max_w \frac{1}{2}\|\nabla f(w)\|_2^2 \quad \text{s.t.} \quad f(w) \le f(w_k).$$

- If $\nabla f(w_k) \neq 0$, then the KKT conditions are necessary for w_k^+ to be a local maximum:

$$\nabla^2 f(w_k^+)\nabla f(w_k^+) - \lambda_k \nabla f(w_k^+) = 0.$$

Level Set Teleportation: Newton's Method

$$w_k^+ \in \arg\max_w \frac{1}{2}\|\nabla f(w)\|_2^2 \quad \text{s.t.} \quad f(w) \le f(w_k).$$

- If $\nabla f(w_k) \neq 0$, then the KKT conditions are necessary for w_k^+ to be a local maximum:

$$\nabla^2 f(w_k^+)\nabla f(w_k^+) - \lambda_k \nabla f(w_k^+) = 0.$$

- λ_k is an eigenvalue of $\nabla^2 f(w_k^+)$ and if $\lambda_k \neq 0$, then

$$\nabla f(w_k^+) = \lambda_k \left[\nabla^2 f(w_k^+)\right]^\dagger \nabla f(w_k^+).$$

Level Set Teleportation: Newton's Method

$$w_k^+ \in \arg\max_w \frac{1}{2}\|\nabla f(w)\|_2^2 \quad \text{s.t.} \quad f(w) \le f(w_k).$$

- If $\nabla f(w_k) \neq 0$, then the KKT conditions are necessary for w_k^+ to be a local maximum:

$$\nabla^2 f(w_k^+)\nabla f(w_k^+) - \lambda_k \nabla f(w_k^+) = 0.$$

- λ_k is an eigenvalue of $\nabla^2 f(w_k^+)$ and if $\lambda_k \neq 0$, then

$$\nabla f(w_k^+) = \lambda_k \left[\nabla^2 f(w_k^+)\right]^\dagger \nabla f(w_k^+).$$

- The gradient direction is the Newton direction with scale λ_k!

Level Set Teleportation: Strong Convexity

Strong Convexity: f is μ-SC if for all $x, y \in \mathbb{R}^d$,

$$f(y) \geq f(x) + \langle \nabla f(x), y - x \rangle + \frac{\mu}{2} \|y - x\|_2^2.$$

Level Set Teleportation: Strong Convexity

Strong Convexity: f is μ-SC if for all $x, y \in \mathbb{R}^d$,

$$f(y) \geq f(x) + \langle \nabla f(x), y - x \rangle + \frac{\mu}{2} \|y - x\|_2^2.$$

Ensures $\nabla^2 f(w_k)$ is P.D. and $\lambda_k > 0$.

Level Set Teleportation: Strong Convexity

Strong Convexity: f is μ-SC if for all $x, y \in \mathbb{R}^d$,

$$f(y) \geq f(x) + \langle \nabla f(x), y - x \rangle + \frac{\mu}{2}\|y - x\|_2^2.$$

Ensures $\nabla^2 f(w_k)$ is P.D. and $\lambda_k > 0$.

Zhao et al. [Zha+23a] use this to analyze GD with teleportation.

Level Set Teleportation: Strong Convexity

Strong Convexity: f is μ-SC if for all $x, y \in \mathbb{R}^d$,

$$f(y) \geq f(x) + \langle \nabla f(x), y - x \rangle + \frac{\mu}{2}\|y - x\|_2^2.$$

Ensures $\nabla^2 f(w_k)$ is P.D. and $\lambda_k > 0$.

Zhao et al. [Zha+23a] use this to analyze GD with teleportation.

- This approach leads to standard, super-linear rates using Newton-type analyses.

Level Set Teleportation: Strong Convexity

Strong Convexity: f is μ-SC if for all $x, y \in \mathbb{R}^d$,

$$f(y) \geq f(x) + \langle \nabla f(x), y - x \rangle + \frac{\mu}{2}\|y - x\|_2^2.$$

Ensures $\nabla^2 f(w_k)$ is P.D. and $\lambda_k > 0$.

Zhao et al. [Zha+23a] use this to analyze GD with teleportation.

- This approach leads to standard, super-linear rates using Newton-type analyses.
- But it requires teleporting before every iteration of GD and strong convexity is key.

Level Set Teleportation: More Problems

And there are some more problems. . .

Level Set Teleportation: More Problems

And there are some more problems. . .

1. Teleportation can blow-up the distance to a minimizer,

$$\|w_k^+ - w^*\|_2 \geq C\|w_k - w^*\|.$$

Level Set Teleportation: More Problems

And there are some more problems...

1. Teleportation can blow-up the distance to a minimizer,

$$\|w_k^+ - w^*\|_2 \geq C\|w_k - w^*\|.$$

   - Breaks standard proof techniques for non-strongly convex f.

Level Set Teleportation: More Problems

And there are some more problems...

1. Teleportation can blow-up the distance to a minimizer,

$$\|w_k^+ - w^*\|_2 \geq C\|w_k - w^*\|.$$

   - Breaks standard proof techniques for non-strongly convex f.

2. If f is non-strongly convex, then $\nabla^2 f(w_k)$ can be positive semi-definite and $\lambda_k = 0$ may happen.

Level Set Teleportation: More Problems

And there are some more problems. . .

1. Teleportation can blow-up the distance to a minimizer,

$$\|w_k^+ - w^*\|_2 \geq C\|w_k - w^*\|.$$

   - Breaks standard proof techniques for non-strongly convex f.

2. If f is non-strongly convex, then $\nabla^2 f(w_k)$ can be positive semi-definite and $\lambda_k = 0$ may happen.

   - Breaks the connection to Newton's method.

Level Set Teleportation: More Problems

And there are some more problems...

1. Teleportation can blow-up the distance to a minimizer,

$$\|w_k^+ - w^*\|_2 \geq C\|w_k - w^*\|.$$

   - Breaks standard proof techniques for non-strongly convex f.

2. If f is non-strongly convex, then $\nabla^2 f(w_k)$ can be positive semi-definite and $\lambda_k = 0$ may happen.

   - Breaks the connection to Newton's method.

3. No efficient algorithms for teleporting in practice!

Analysis and Algorithms

Convergence Analysis: Convex Functions

Goal: fast rates for GD with intermittent teleportation.

Convergence Analysis: Convex Functions

Goal: fast rates for GD with intermittent teleportation.

The key to faster rates is the KKT stationarity condition,

$$\nabla f(w_k^+) = \lambda_k \left[\nabla^2 f(w_k^+)\right]^\dagger \nabla f(w_k^+).$$

Convergence Analysis: Convex Functions

Goal: fast rates for GD with intermittent teleportation.

The key to faster rates is the KKT stationarity condition,

$$\nabla f(w_k^+) = \lambda_k \left[\nabla^2 f(w_k^+)\right]^\dagger \nabla f(w_k^+).$$

- But, we can't use this because $\lambda_k = 0$ may hold...

Convergence Analysis: Convex Functions

Goal: fast rates for GD with intermittent teleportation.

The key to faster rates is the KKT stationarity condition,

$$\nabla f(w_k^+) = \lambda_k \left[\nabla^2 f(w_k^+)\right]^\dagger \nabla f(w_k^+).$$

- But, we can't use this because $\lambda_k = 0$ may hold...
- Even worse, $\|w_k^+ - w^*\|_2^2 > C\|w_k - w^*\|_2^2$ breaks the standard GD recursion:

$$\begin{aligned}\|w_{k+1} - w^*\|_2^2 = \|w_k^+ - w^*\|_2^2 \\ - 2\eta_k \left\langle \nabla f(w_k^+), w_k^+ - w^* \right\rangle + \eta_k^2 \|\nabla f(w_k^+)\|_2^2,\end{aligned}$$

Convergence Analysis: Convex Functions

Solution: Teleportation is non-expansive in function values,

$$f(w_{k+1}) \leq f(w_k^+) \leq f(w_k).$$

Convergence Analysis: Convex Functions

Solution: Teleportation is non-expansive in function values,

$$f(w_{k+1}) \leq f(w_k^+) \leq f(w_k).$$

As a result, all iterates remain in the initial sub-level set

$$\mathcal{S}_0 := \{w : f(w) \leq f(w_0)\}.$$

Convergence Analysis: Convex Functions

Solution: Teleportation is non-expansive in function values,

$$f(w_{k+1}) \leq f(w_k^+) \leq f(w_k).$$

As a result, all iterates remain in the initial sub-level set

$$\mathcal{S}_0 := \{w : f(w) \leq f(w_0)\}.$$

Theorem (Informal)

Let $R = \sup\{\|w - w^*\|_2 : w \in \mathcal{S}_0\}$. *If* f *is* L*-smooth and convex, then GD with* $\eta < 2/L$ *and teleportation schedule* $\mathcal{T}$ *satisfies,*

$$f(w_K) - f(w^*) \leq \frac{2R^2}{K\eta(2 - L\eta)}.$$

Moreover, there exists a function for which the convergence of GD with and without teleportation are identical.

Convergence: A Negative Tightness Result

Denote $\delta_K = f(w_K) - f(w^*)$. Comparing to standard GD [Bub15],

$$\underbrace{\delta_K \leq \frac{2R^2}{K\eta\,(2 - L\eta)}}_{\text{With Teleportation}} \quad \textbf{vs} \quad \underbrace{\delta_K \leq \frac{2\|w_0 - w^*\|_2^2}{K\eta\,(2 - L\eta)}}_{\text{Without Teleportation}}.$$

Convergence: A Negative Tightness Result

Denote $\delta_K = f(w_K) - f(w^*)$. Comparing to standard GD [Bub15],

$$\underbrace{\delta_K \leq \frac{2R^2}{K\eta\,(2-L\eta)}}_{\text{With Teleportation}} \quad \textbf{vs} \quad \underbrace{\delta_K \leq \frac{2\|w_0 - w^*\|_2^2}{K\eta\,(2-L\eta)}}_{\text{Without Teleportation}}.$$

Unfortunately, the dependence on the diameter is tight.

Theorem (Informal)

There exists an L-smooth and convex function such that teleporting from the initialization guarantees,

$$\|w_0^+ - w^*\|_2 \geq R/4.$$

Convergence: Hessian Stability

We need the connection to Newton's method to prove fast rates.

Convergence: Hessian Stability

We need the connection to Newton's method to prove fast rates.

- $\nabla^2 f(w)$ needs to be positive definite and well-behaved.

Convergence: Hessian Stability

We need the connection to Newton's method to prove fast rates.

- $\nabla^2 f(w)$ needs to be positive definite and well-behaved.

Definition (Hessian Stability)

We say f has $(\tilde{L}, \tilde{\mu})$-stable Hessian over $\mathcal{Q} \subseteq \mathbb{R}^d$ if for every $x, y \in \mathcal{Q}$, $\nabla^2 f(x)(y - x) \neq 0$ and,

$$f(y) \leq f(x) + \langle \nabla f(x), y - x \rangle + \frac{\tilde{L}}{2} \|y - x\|_{\nabla^2 f(x)}^2,$$

$$f(y) \geq f(x) + \langle \nabla f(x), y - x \rangle + \frac{\tilde{\mu}}{2} \|y - x\|_{\nabla^2 f(x)}^2.$$

Convergence: Hessian Stability

We need the connection to Newton's method to prove fast rates.

- $\nabla^2 f(w)$ needs to be positive definite and well-behaved.

Definition (Hessian Stability)

We say f has $(\tilde{L}, \tilde{\mu})$-stable Hessian over $\mathcal{Q} \subseteq \mathbb{R}^d$ if for every $x, y \in \mathcal{Q}$, $\nabla^2 f(x)(y-x) \neq 0$ and,

$$f(y) \leq f(x) + \langle \nabla f(x), y-x \rangle + \frac{\tilde{L}}{2} \|y-x\|^2_{\nabla^2 f(x)},$$

$$f(y) \geq f(x) + \langle \nabla f(x), y-x \rangle + \frac{\tilde{\mu}}{2} \|y-x\|^2_{\nabla^2 f(x)}.$$

Holds for practical problems, including logistic regression [Bac10].

Convergence: Fast Rates under Hessian Stability

Recall that $\delta_k = f(w_k) - f(w^*)$ is the optimality gap.

Convergence: Fast Rates under Hessian Stability

Recall that $\delta_k = f(w_k) - f(w^*)$ is the optimality gap.

Theorem (Informal)

Suppose f is L-smooth, convex, and satisfies Hessian stability on $\mathcal{S}_0$. Let $M = K - |\mathcal{T}|$. Then GD with line-search satisfies,

$$\delta_K \leq \frac{2R^2 L}{M + 2R^2 L \sum_{k \in \mathcal{B}} \left[\left(\frac{\tilde{L}}{\tilde{L} - \tilde{\mu}} \right)^{b_k} - 1 \right] \frac{1}{\delta_{k-1}}}.$$

Convergence: Fast Rates under Hessian Stability

Recall that $\delta_k = f(w_k) - f(w^*)$ is the optimality gap.

Theorem (Informal)

Suppose f is L-smooth, convex, and satisfies Hessian stability on $\mathcal{S}_0$. Let $M = K - |\mathcal{T}|$. Then GD with line-search satisfies,

$$\delta_K \leq \frac{2R^2 L}{M + 2R^2 L \sum_{k \in \mathcal{B}} \left[\left(\frac{\tilde{L}}{\tilde{L} - \tilde{\mu}} \right)^{b_k} - 1 \right] \frac{1}{\delta_{k-1}}}.$$

Convergence is faster than GD when δ_k is small!

Convergence: Simplified Rates

- An alternative proof technique makes this rate explicit.

Convergence: Simplified Rates

- An alternative proof technique makes this rate explicit.
- Consider teleporting every other iteration: $\mathcal{T} = \{1, 3, 5, \ldots\}$.

Convergence: Simplified Rates

- An alternative proof technique makes this rate explicit.
- Consider teleporting every other iteration: $\mathcal{T} = \{1, 3, 5, \ldots\}$.

Theorem

Suppose f is L-smooth, convex, and satisfies Hessian stability on $\mathcal{S}_0$. Then GD with line-search satisfies,

$$\delta_K \leq \frac{2R^2 L(\tilde{L} - \tilde{\mu})}{\tilde{\mu}\left[\left(\frac{\tilde{L}}{\tilde{L} - \tilde{\mu}}\right)^{K/2} - 1\right]} \approx 2R^2 L\tilde{C}\left(1 - \frac{\tilde{\mu}}{\tilde{L}}\right)^{K/2}.$$

Convergence: Simplified Rates

- An alternative proof technique makes this rate explicit.
- Consider teleporting every other iteration: $\mathcal{T} = \{1, 3, 5, \ldots\}$.

Theorem

Suppose f is L-smooth, convex, and satisfies Hessian stability on $\mathcal{S}_0$. Then GD with line-search satisfies,

$$\delta_K \leq \frac{2R^2 L(\tilde{L} - \tilde{\mu})}{\tilde{\mu}\left[\left(\frac{\tilde{L}}{\tilde{L}-\tilde{\mu}}\right)^{K/2} - 1\right]} \approx 2R^2 L\tilde{C}\left(1 - \frac{\tilde{\mu}}{\tilde{L}}\right)^{K/2}.$$

This is a linear rate without strong convexity!

Level Set Teleportation: Towards an Algorithm

$$w_k^+ \in \arg\max_{w} \frac{1}{2}\|\nabla f(w)\|_2^2 \quad \text{s.t.} \quad f(w) \leq f(w_k).$$

Level Set Teleportation: Towards an Algorithm

$$w_k^+ \in \arg\max_{w} \frac{1}{2}\|\nabla f(w)\|_2^2 \quad \text{s.t.} \quad f(w) \leq f(w_k).$$

Restrictions on an efficient teleportation algorithm.

- The derivative of the gradient norm is a Hessian-vector product: $\nabla^2 f(x) \nabla f(x)$.

Level Set Teleportation: Towards an Algorithm

$$w_k^+ \in \arg\max_{w} \frac{1}{2}\|\nabla f(w)\|_2^2 \quad \text{s.t.} \quad f(w) \leq f(w_k).$$

Restrictions on an efficient teleportation algorithm.

- The derivative of the gradient norm is a Hessian-vector product: $\nabla^2 f(x) \nabla f(x)$.
- We can't use generic second-order solvers, since computing the Hessian requires third-order derivatives.

Level Set Teleportation: Towards an Algorithm

$$w_k^+ \in \arg\max_{w} \frac{1}{2}\|\nabla f(w)\|_2^2 \quad \text{s.t.} \quad f(w) \leq f(w_k).$$

Restrictions on an efficient teleportation algorithm.

- The derivative of the gradient norm is a Hessian-vector product: $\nabla^2 f(x) \nabla f(x)$.
- We can't use generic second-order solvers, since computing the Hessian requires third-order derivatives.
- The algorithm must be parameter-free because teleportation is only a sub-routine of GD.

Level Set Teleportation: Towards an Algorithm

$$w_k^+ \in \arg\max_{w} \frac{1}{2}\|\nabla f(w)\|_2^2 \quad \text{s.t.} \quad f(w) \leq f(w_k).$$

Restrictions on an efficient teleportation algorithm.

- The derivative of the gradient norm is a Hessian-vector product: $\nabla^2 f(x) \nabla f(x)$.
- We can't use generic second-order solvers, since computing the Hessian requires third-order derivatives.
- The algorithm must be parameter-free because teleportation is only a sub-routine of GD.

These constraints suggest first-order methods.

Level Set Teleportation: Practical Teleportation

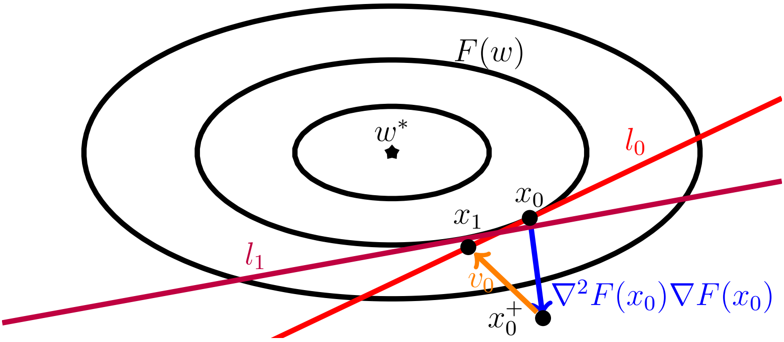
We can't project onto $\mathcal{L}_k := \{w : f(w) = f(w_k)\}$, but we can project onto the linearization at x_k:

$$l_k = \{w : f(x_k) + \langle \nabla f(x_k), w - x_k \rangle = f(w_k)\}$$

Level Set Teleportation: Practical Teleportation

We can't project onto $\mathcal{L}_k := \{w : f(w) = f(w_k)\}$, but we can project onto the linearization at x_k:

$$l_k = \{w : f(x_k) + \langle \nabla f(x_k), w - x_k \rangle = f(w_k)\}$$



Level Set Teleportation: Practical Teleportation

Algorithm Sub-level Set Teleportation

$x_0 \leftarrow w_k$
$q_0 \leftarrow \nabla^2 f(x_0)\nabla f(x_0)$
while $\|\mathbf{P}_t q_t\|_2 > \epsilon$ or $f(x_k) - f(w_k) > \delta$ **do**
  $g_t \leftarrow \|\nabla f(x_k)\|_2^2$
  $v_t \leftarrow -\left(\rho \langle q_t, \nabla f(x_k)\rangle + f(x_k) - f(w_k)\right)_+ \nabla f(x_k)$
  $x_{k+1} \leftarrow x_k + (\rho \cdot q_t + v_t)/g_t$
  while $\phi_{\gamma_t}(x_{k+1}) > \frac{1}{2}g_t + (\langle q_t, v_t\rangle - \rho\|q_t\|_2^2)/g_t$ **do**
    $\rho \leftarrow \rho/2$
    $v_t \leftarrow -(\rho \langle q_t,\nabla f(x_k)\rangle + f(x_k) - f(w_k))_+ \nabla f(x_k)$
    $x_{k+1} \leftarrow x_k + (\rho \cdot q_t + v_t)/g_t$
  end while
  $q_{t+1} \leftarrow \nabla^2 f(x_k)\nabla f(x_k)$
  $t \leftarrow t + 1$
end while
Output: x_{k+1}

Experiments

Experiments: Evaluating the Algorithm

Consider teleportation for a non-convex neural network.

Experiments: Evaluating the Algorithm

Consider teleportation for a non-convex neural network.

- We consider a two-layer MLP with 50 hidden units and soft-plus activations.

Experiments: Evaluating the Algorithm

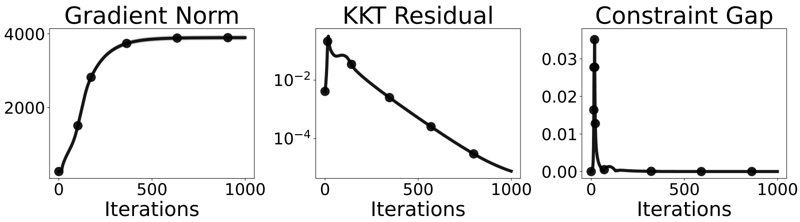
Consider teleportation for a non-convex neural network.

- We consider a two-layer MLP with 50 hidden units and soft-plus activations.
- Dataset is `MNIST`.

Experiments: Evaluating the Algorithm

Consider teleportation for a non-convex neural network.

- We consider a two-layer MLP with 50 hidden units and soft-plus activations.
- Dataset is `MNIST`.



Experiments: Performance Profile

We generate 120 problems from the UCI repository [AN07].

Experiments: Performance Profile

We generate 120 problems from the UCI repository [AN07].

- Solid lines indicate methods with teleportation.

Experiments: Performance Profile

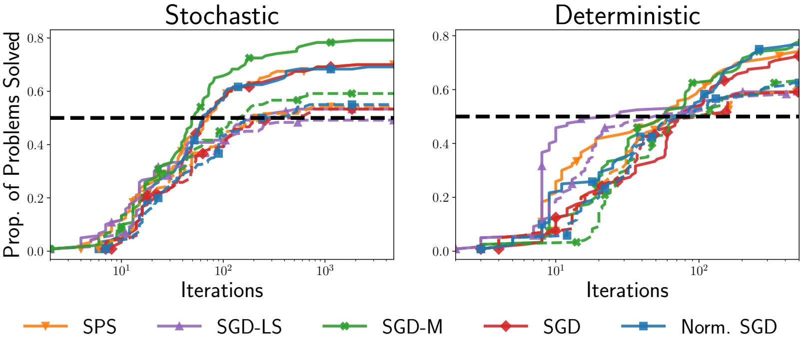
We generate 120 problems from the UCI repository [AN07].

- Solid lines indicate methods with teleportation.
- Dashed lines are the same methods without teleportation.

Experiments: Performance Profile

We generate 120 problems from the UCI repository [AN07].

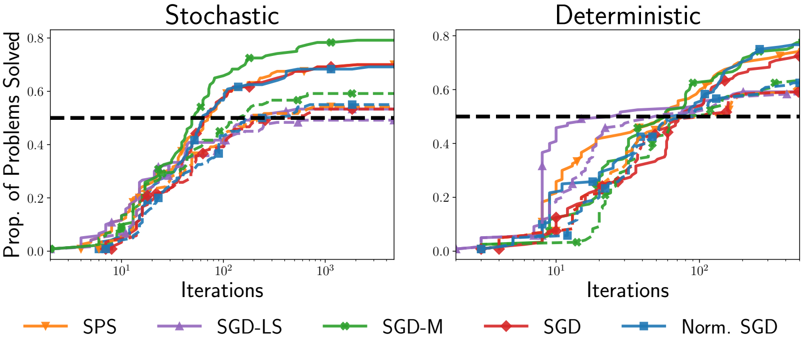
- Solid lines indicate methods with teleportation.
- Dashed lines are the same methods without teleportation.



Experiments: Performance Profile

We generate 120 problems from the UCI repository [AN07].

- Solid lines indicate methods with teleportation.
- Dashed lines are the same methods without teleportation.



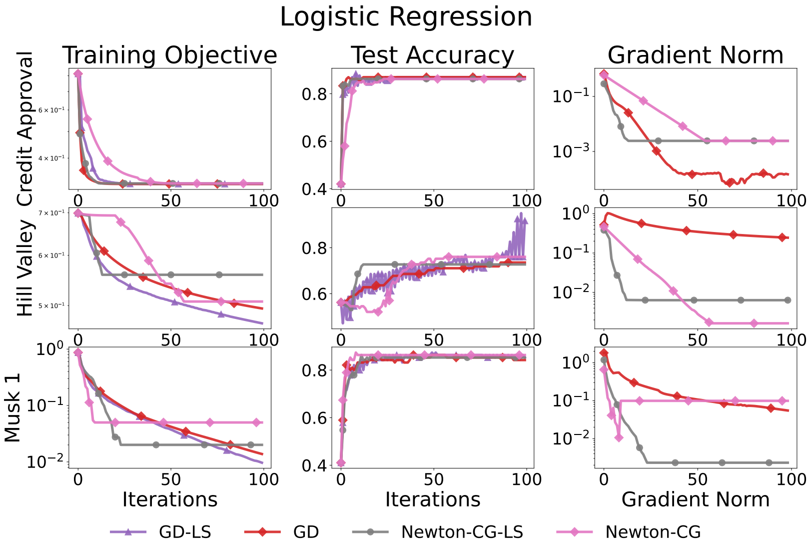
Teleportation uniformly improves speed and success rates!

Experiments: Convex Newton

Teleporting at every iteration behaves like Newton's method.

Experiments: Convex Newton

Teleporting at every iteration behaves like Newton's method.

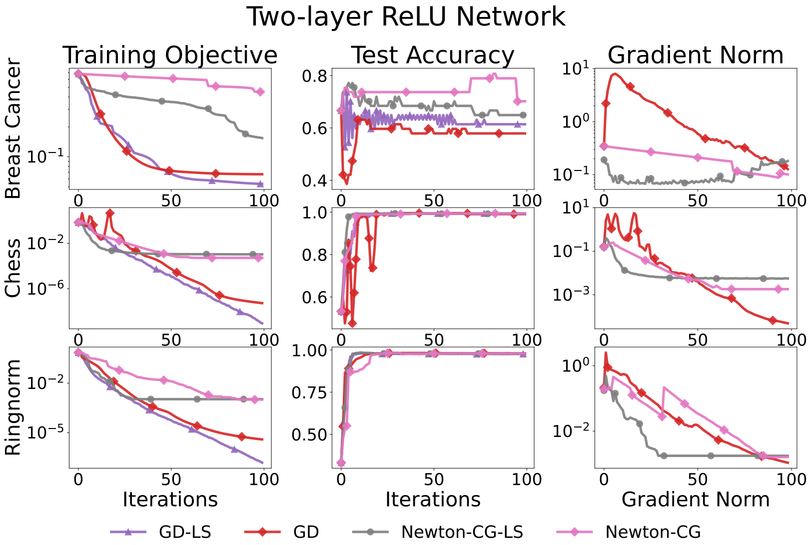


Experiments: Convex Newton

But teleportation still works for non-convex problems!

Experiments: Convex Newton

But teleportation still works for non-convex problems!



Questions?

References I

[AJ21] Marco Armenta and Pierre-Marc Jodoin. "The representation theory of neural networks". In: Mathematics 9.24 (2021), p. 3216.

[AN07] Arthur Asuncion and David Newman. UCI machine learning repository. 2007.

[Arm+20] Marco Armenta et al. "Neural Teleportation". In: CoRR abs/2012.01118 (2020).

[Bac10] Francis Bach. "Self-concordant analysis for logistic regression". In: Electronic Journal of Statistics 4 (2010), pp. 384–414.

[Bub15] Sébastien Bubeck. "Convex Optimization: Algorithms and Complexity". In: Found. Trends Mach. Learn. 8.3-4 (2015), pp. 231–357.

References II

[FA00] Kenji Fukumizu and Shun-ichi Amari. "Local minima and plateaus in hierarchical structures of multilayer perceptrons". In: Neural networks 13.3 (2000), pp. 317–327.

[KNS16] Hamed Karimi, Julie Nutini, and Mark Schmidt. "Linear Convergence of Gradient and Proximal-Gradient Methods Under the Polyak-Łojasiewicz Condition". In: Machine Learning and Knowledge Discovery in Databases - European C Vol. 9851. Lecture Notes in Computer Science. 2016, pp. 795–811.

[Zha+23a] Bo Zhao et al. "Improving Convergence and Generalization Using Parameter Symmetries". In: CoRR abs/2305.13404 (2023).

[Zha+23b] Bo Zhao et al. Symmetry Teleportation for Accelerated Optimization. 2023. arXiv: 2205.10637 [cs.LG].