Towards Provable, Efficient and Robust Data-Driven Optimization

Junqi Tang

Joint work with: **Hong Ye Tan**, Subhadip Mukherjee, Andreas Hauptmann, Carola-Bibiane Schönlieb

School of Mathematics, University of Birmingham, UK

Workshop on Recent Advances in Iterative Reconstruction, 22 May, 2023

Learning to optimize (L2O)

This talk is based on our works:

- Data-Driven Mirror Descent with Input-Convex Neural Networks. SIAM Journal on Mathematics of Data Science (SIMODS), 2023
- Robust Data-Driven Accelerated Mirror Descent. ICASSP 2023

In this line of works, we propose new L2O paradigms based on mirror descent.

Introduction

- Large-scale optimization problems $\min_{x \in \mathcal{X}} f(x)$ are ubiquitous in machine learning, data science, computational imaging, $\cdots$

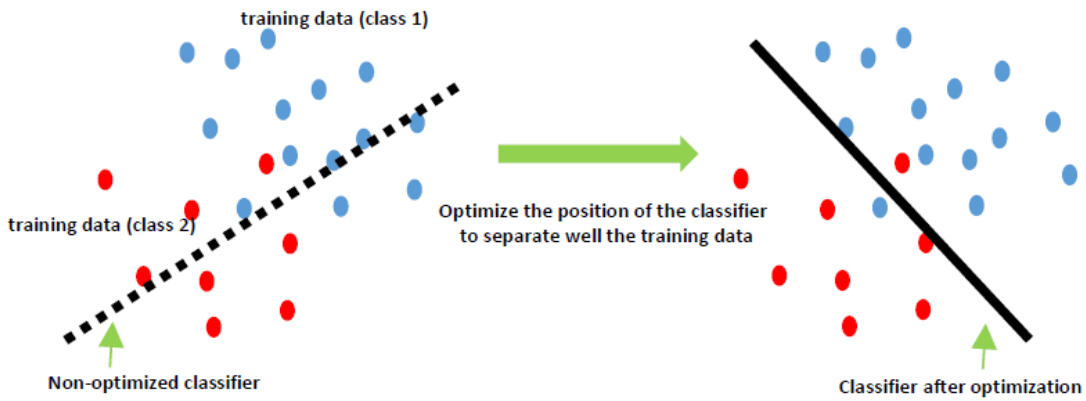
Introduction

Example: Empirical risk minimization in machine learning

- Training a prediction function via empirical risk minimization

$$x^\star \in \arg\min_{x\in\mathbb{R}^d} f(x) := \underbrace{\frac{1}{n}\sum_{i=1}^{n} l(b_i, h(a_i, x))}_{\text{Data fidelity}} + \underbrace{\lambda g(x)}_{\text{regularization}} ,$$

(SVM , kernel methods, deep neural networks, etc):



Introduction

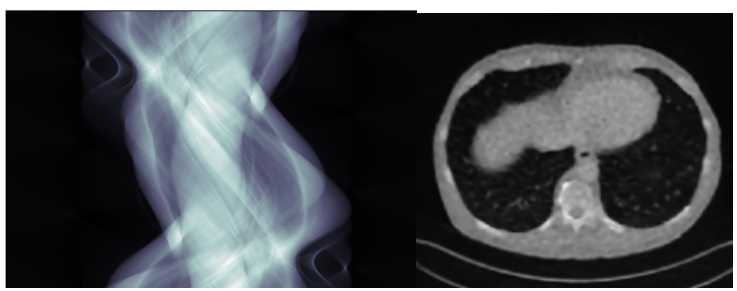
Example in inverse problems

- **Tomographic medical imaging** (CT, MRI, PET...):

$$b = Ax^{\dagger} + noise,$$

estimate $x^{\dagger}$ via solving:

$$x^{\star} \in \arg\min_{x \in \mathbb{R}^d} f(x) := \underbrace{\frac{1}{n}\sum_{i=1}^{n} l(b_i, a_i^T x)}_{\text{Data fidelity}} + \underbrace{\lambda g(x)}_{\text{regularization}} ,$$



measurements (b) Solution ($x^{\star}$)

Introduction

Large-scale optimization

- The number of data-sample n and dimension d can be huge in modern data science applications.
  – leading to significant computational challenges for optimization algorithms!

Introduction

Optimal algorithms for convex composite finite-sum optimization

- Recall our generic objective: $f(x) := \frac{1}{n}\sum_{i=1}^{n} l_i(x) + \lambda g(x)$
- $F(x)$ is convex and each f_i has L-lipschitz continous gradient.

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

- **Optimal algorithms** – SGD with variance-reduction + momentum acceleration[1] achieves worse-case optimal convergence.

> *To achieve* $\mathbb{E}[f(x^t)] - f(x^\star) \leq \epsilon$, $O(n + \sqrt{\frac{nL}{\epsilon}})$ *gradient evaluation is needed.*

Matching the lower bound $\Omega(n + \sqrt{\frac{nL}{\epsilon}})$ [Woodworth&Srebro, NeurIPS'16]

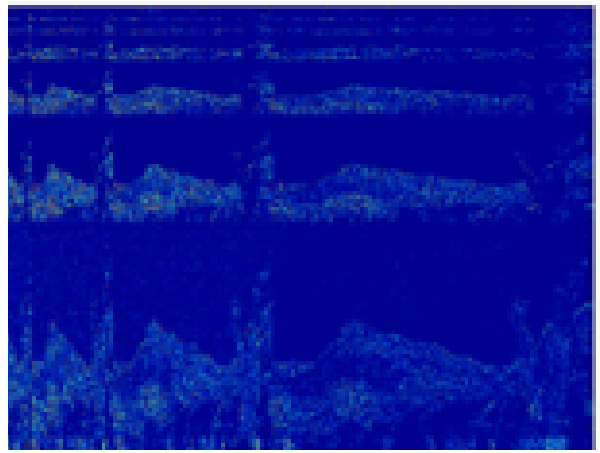
[1] Representative examples of optimal SGD methods: Katyusha [Allen-Zhu, STOC'17], MiG [Zhou et al, ICML'18],Varag [Lan et al, NeurIPS'19]...

What optimization algorithms have missed out...

Real-world data is highly structured!!!!!

What optimization algorithms have missed out...

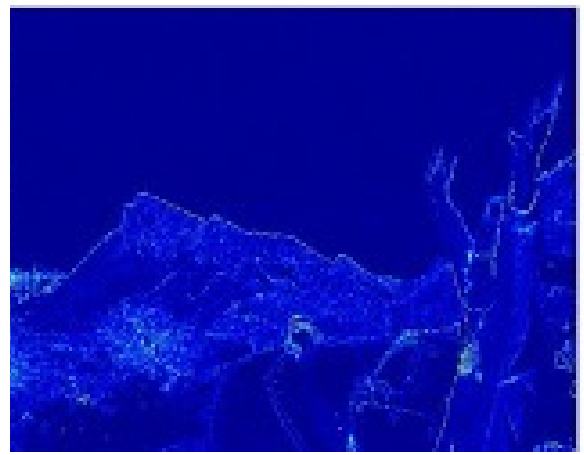
Real-world data is highly structured!!!!!



If we take a wavelet transform, we can see that at least 90% of coefficients are nearly zeros.

What optimization algorithms have missed out...

Real-world data is highly structured!!!!!

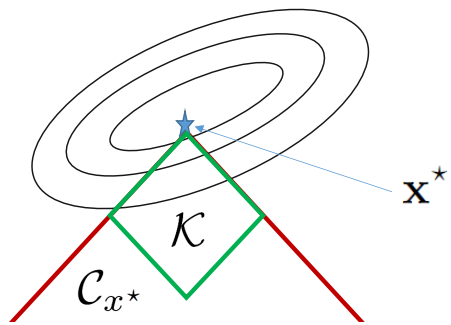


If we take the gradient of an image, we can see that around 95% of coefficients are nearly zeros.

The Limitations of Classical Algorithmic Design Paradigms

The current paradigms in large-scale optimization focuses on generic algorithms for wide classes,

- ignoring the intrinsic low-dimensional structure of the problem



- ignoring the data structure/distribution in specific applications

– may lead to suboptimal practical performances

Hand-crafting specialized algorithms for every narrow subclass is impractical.

Learning to optimize (L2O)

- Classical paradigm: given a generic class of optimization problem, design an efficient algorithm.
- L2O paradigm: given random instances of problems from a target task distribution, learn a solution algorithm to solve novel random instances from the same task distribution.

Objective: Combine machine learning with optimization to obtain better solutions faster, with provable convergence rates!

Background on mirror descent (MD)

Gradient descent (GD): $x_{k+1} = x_k - t_k \nabla f(x_k)$

$$x_{k+1} = \arg\min_{x \in \mathbb{R}^n} \left\{ \langle x, \nabla f(x_k) \rangle + \frac{1}{2t_k} \|x - x_k\|_2^2 \right\}$$

Mirror descent (MD):

$$x_{k+1} = \arg\min_{x \in \mathbb{R}^n} \left\{ \langle x, \nabla f(x_k) \rangle + \frac{1}{t_k} B_\Phi(x, x_k) \right\}$$

- Φ is strongly convex, continuously differentiable (mirror potential).
- Bregman distance: $B_\Phi(x, y) = \Phi(x) - \Phi(y) - \langle \nabla \Phi(y), x - y \rangle$.
- Convex conjugate $\Phi^*(u) = \sup_{x \in \mathcal{X}} \{\langle u, x \rangle - \Phi(x)\}$, satisfies $\nabla \Phi^* = (\nabla \Phi)^{-1}$.
- MD update: $y_k = \nabla \Phi(x_k) - t_k \nabla f(x_k),\ x_{k+1} = \nabla \Phi^*(y_k)$.

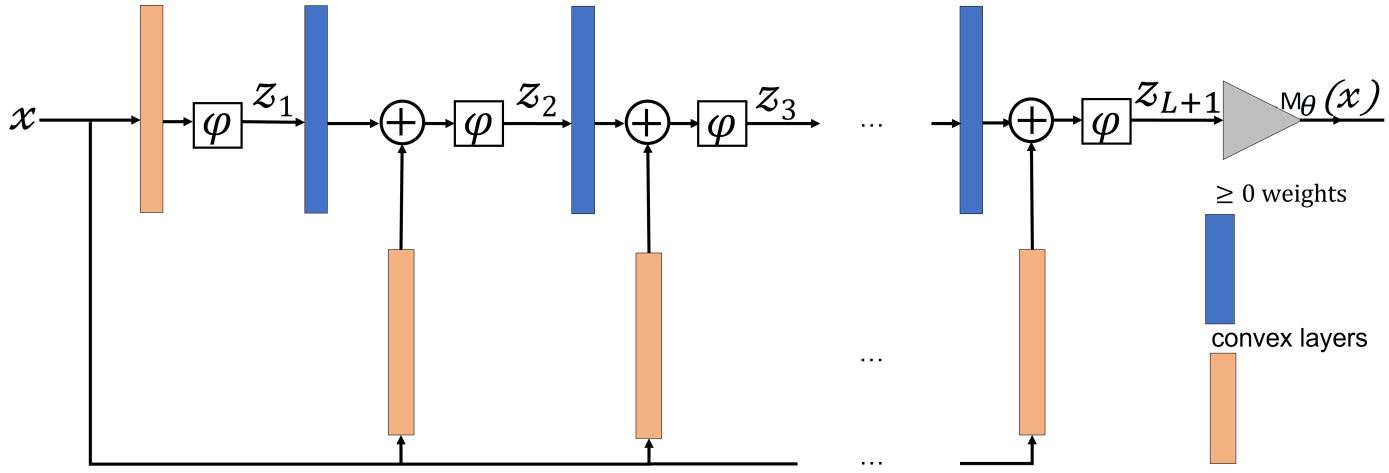
Parameterizing the mirror potential Φ

- Parameterize Φ using an input-convex neural network (ICNN): M_θ
- Architecture of M_θ:

$$
\begin{aligned}
z_0(x) &= 0 \\
z_{i+1}(x) &= \varphi_i\left(B_i\left(z_i(x)\right) + W_i(x) + b_i\right), i = 0, 1, \cdots, L-1 \\
M_\theta(x) &= \texttt{AvgPool}\left(z_L(x)\right), \ \theta = (B_i, W_i, b_i)_{i=0}^{L-1}
\end{aligned}
$$

- $W_i : x \mapsto W_i(x)$ convex, B_i: `conv2D` layers with ≥ 0 weights
- φ_i: point-wise convex and monotonically non-decreasing (e.g., `relu`/`leaky_relu`).

ICNN architecture



- The construction is recursive, where each $z_i(x)$ is a vector whose elements are convex functions of the input x.
- The construction uses the following two properties:

1. $\sum_i w_i u_i(x)$ is convex in convex in x if each u_i is convex and $w_i \geq 0$.

2. $\varphi \circ u$ is convex if both u and φ are convex and φ is monotonically-increasing.

Main challenges

- Recall that MD needs both Φ and Φ^*.
- If Φ is modeled using an ICNN, how does one compute/approximate the gradient of Φ^*?
- We use another network M_ϑ to approximate Φ^*, and then enforce $\nabla M_\vartheta^* = (\nabla M_\theta)^{-1}$ using a soft penalty during training.
- No longer have an exact MD algorithm, so can we quantify the regret w.r.t. the approximation error? Yes!

Regret of approximate MD

- Exact MD: $x_{k+1} = \nabla\Phi^* \left(\nabla\Phi(x_k) - t_k \nabla f(x_k)\right)$
- Approximate MD: $\tilde{x}_{k+1} = \widetilde{\nabla\Phi}^* \left(\nabla\Phi(x_k) - t_k \nabla f(x_k)\right)$

Regret Bound for approximate MD: Suppose f is μ-strongly convex and Φ is a mirror potential with strong convexity parameter σ. Let $\{\tilde{x}_k\}_{k=0}^{\infty}$ be some sequence in $\mathcal{X} \subseteq \mathbb{R}^n$, and $\{x_k\}_{k=1}^{\infty}$ be the corresponding exact MD iterates. We have the following regret bound:

$$\sum_{k=1}^{K} t_k (f(\tilde{x}_k) - f(x^*)) \leq B(x^*, \tilde{x}_1) + \sum_{k=1}^{K} \left[\frac{t_k^2}{\sigma} \|\nabla f(\tilde{x}_k)\|_*^2 + \left(\frac{1}{2t_k\mu} + \frac{1}{\sigma}\right) \underbrace{\|\nabla\Phi(\tilde{x}_{k+1}) - \nabla\Phi(x_{k+1})\|_*^2}_{\text{approximation error } \delta_{k+1}} \right]$$

$$\delta_k = \|\nabla\Phi(\tilde{x}_k) - \nabla\Phi(x_k)\|_*^2 = \left\| \left(\nabla\Phi \circ \widetilde{\nabla\Phi}^* - \mathsf{Id} \right)(y_k) \right\|_*^2,$$

where $y_k := \nabla\Phi(x_k) - t_k \nabla f(x_k)$.

Training objective

- Parameterizing the solution operator via unrolling:

$$\tilde{x}_{k+1} = \nabla M_\vartheta^*(\nabla M_\theta(\tilde{x}_k) - t_k \nabla f(\tilde{x}_k)),\ k = 0, 1, 2, \cdots, N-1.$$

- Training loss:

$$L(\theta, \vartheta) = \mathbb{E}_{f \in \mathcal{F}, \tilde{x}_0} \left[\sum_{k=1}^{N} f(\tilde{x}_k) + s_k \left\| (\nabla M_\vartheta^* \circ \nabla M_\theta - \mathsf{Id})(\tilde{x}_k) \right\| \right]$$

  - Two-fold goal: decrease in the target objective + ensure forward-backward consistency.
  - We set $s_k = s$ for all k, and then increase s in every epoch.

A couple of concrete examples

- Support vector machine (SVM):

$$\boxed{\min_{x=(\mathbf{w},b)} \frac{1}{2}\mathbf{w}^\top\mathbf{w} + C\sum_{i\in\mathcal{I}} \max(0, 1 - y_i(\mathbf{w}^\top\phi_i + b))}$$

$$\mathcal{F} = \left\{ f_{\mathcal{I}}(\mathbf{w}, b) = \frac{1}{2}\mathbf{w}^\top\mathbf{w} + C\sum_{i\in\mathcal{I}} \max(0, 1 - y_i(\mathbf{w}^\top\phi_i + b)) \right\}.$$

  - Each instance of f depends on the set $\mathcal{I}$ of feature-target pairs.

- Image Inpainting:

$$\boxed{\min_x \frac{1}{2}\|Ax - y\|_2^2 + \lambda\|\nabla x\|_1}$$

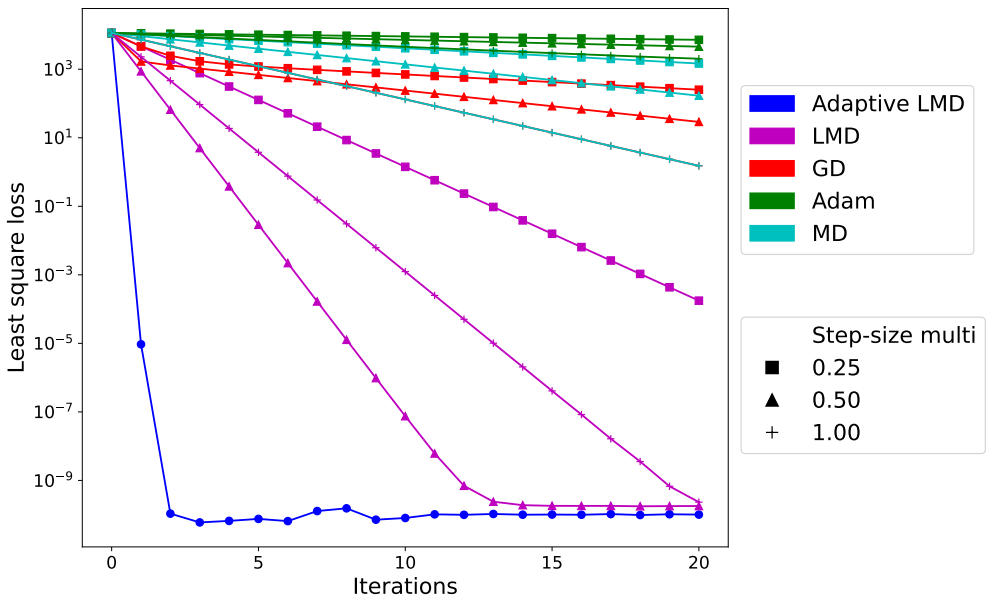
$$\mathcal{F} = \left\{ f(x) = \frac{1}{2}\|Ax - y\|_2^2 + \lambda\|\nabla x\|_1 : \text{noisy images } y \right\}.$$

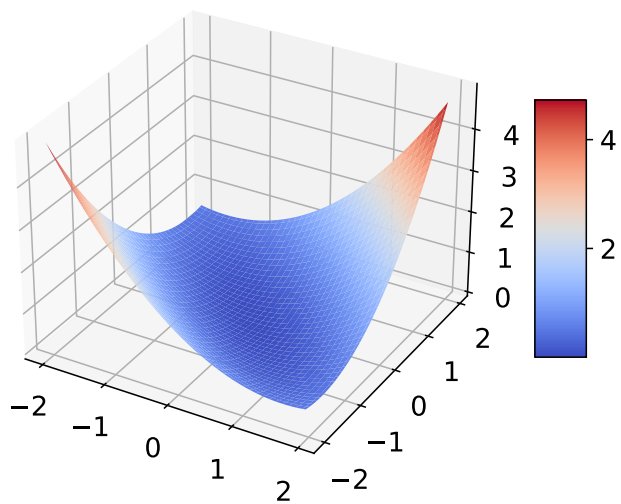
Numerical results

Toy example: least squares in $\mathbb{R}^2$

$$\min_{x\in\mathbb{R}^2} \|Wx-b\|_2^2,\ W=\begin{pmatrix}2 & 1\\ 1 & 2\end{pmatrix}, \mathcal{F}=\{f_b(x)=\|Wx-b\|_2^2 : b\in\mathbb{R}^2\}$$

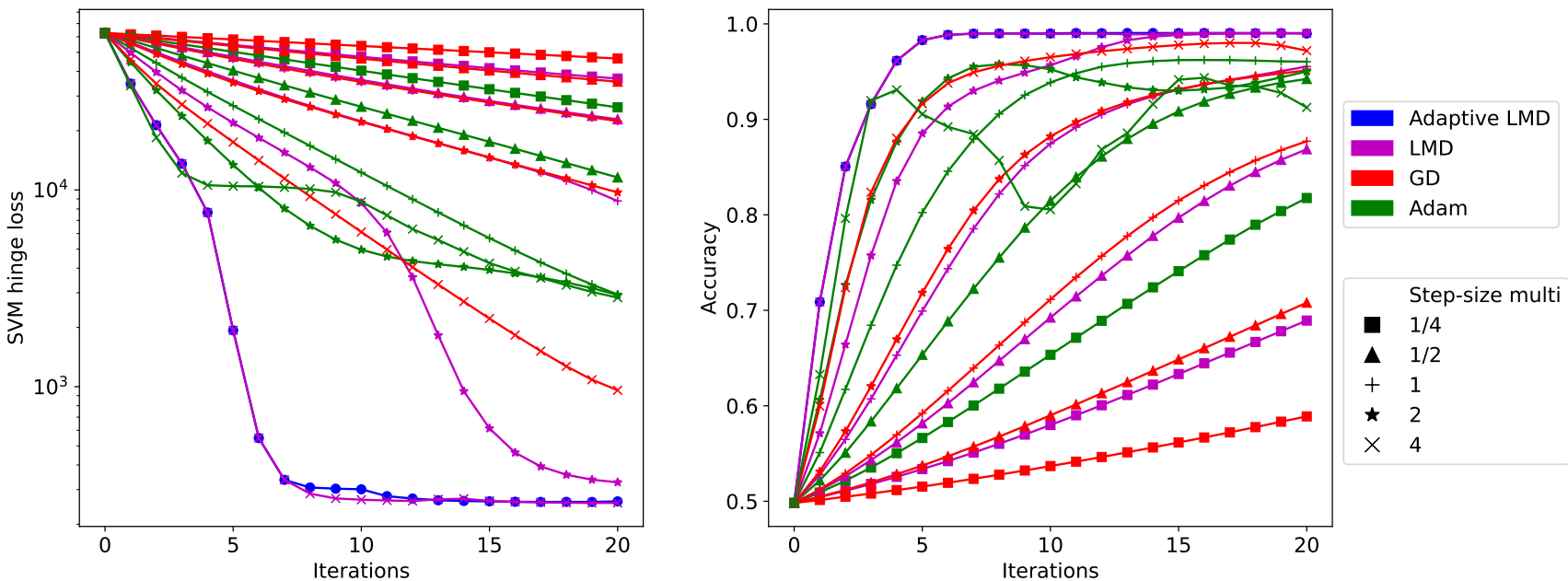
- Theoretically optimal mirror map: $\Phi(x)=\frac{1}{2}x^\top(W^\top W)x$
- Our parameterization: $\Phi(x)=\frac{1}{2}x^\top Ax$, A symmetric PD
- Learned MD (LMD) $\to \begin{pmatrix}0.69 & 0.55\\ 0.55 & 0.69\end{pmatrix}$, nearly $\propto W^\top W=\begin{pmatrix}5 & 4\\ 4 & 5\end{pmatrix}$





SVM training

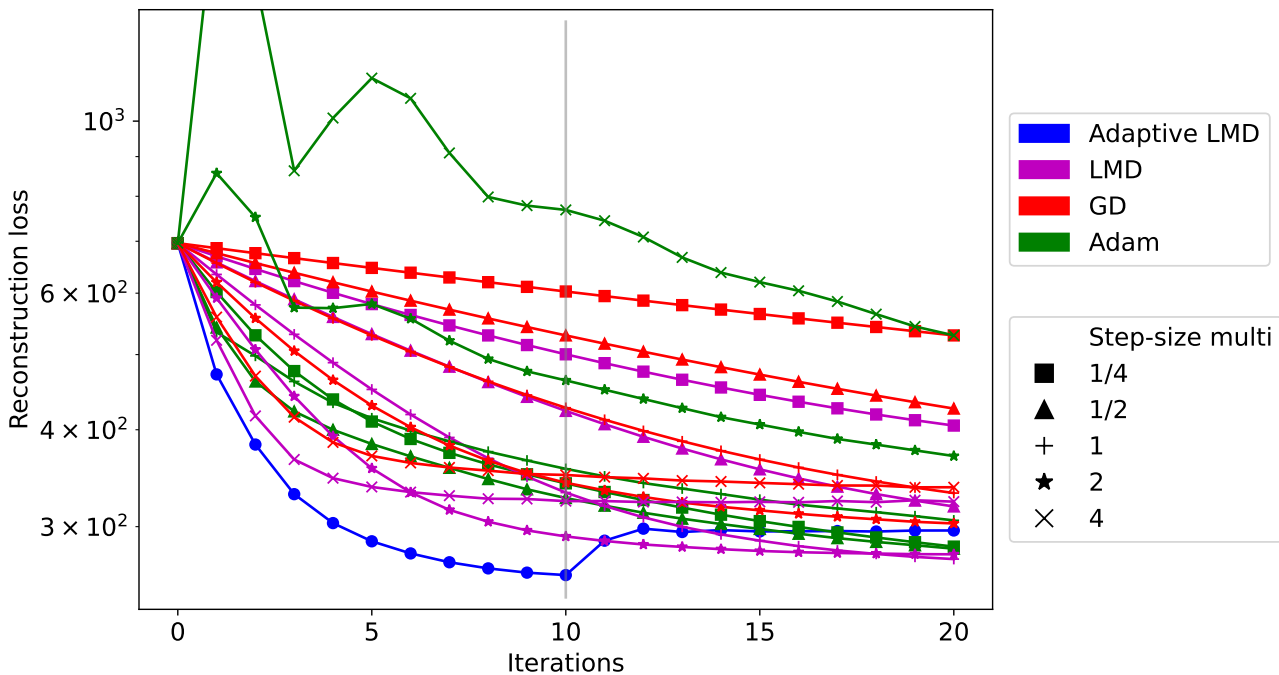
- Data: MNIST, two-class SVM (digits 4 and 9)
- Trained on features $\phi : \mathbb{R}^{28^2} \mapsto \mathbb{R}^{50}$ extracted by a neural net trained with 97% accuracy
- Trained and tested on different folds
- Only 10 iterations are trained and then extended with various step-size multipliers



TV denoising (Gaussian noise)

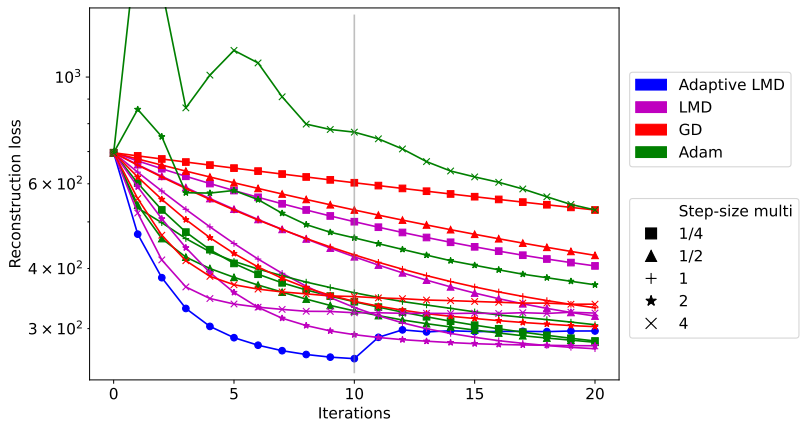
- Data: STL-10 images
- LMD trained with noise-level $\sigma = 0.05$ for 10 iterations

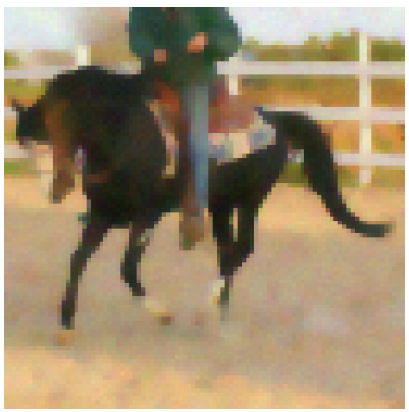
$$\mathcal{F} = \left\{ f(x) = \frac{1}{2}\|x - y\|_2^2 + \lambda \|\nabla x\|_1 : \text{noisy images } y \right\}.$$



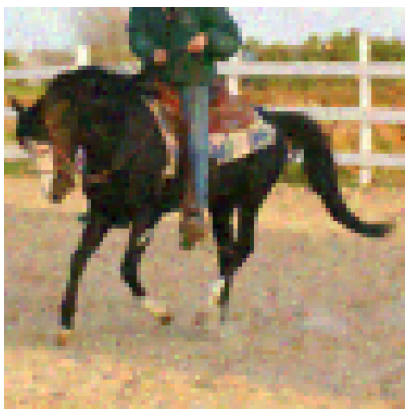
TV denoising (Gaussian noise)

- Data: STL-10 images
- LMD trained with noise-level $\sigma = 0.05$ for 10 iterations

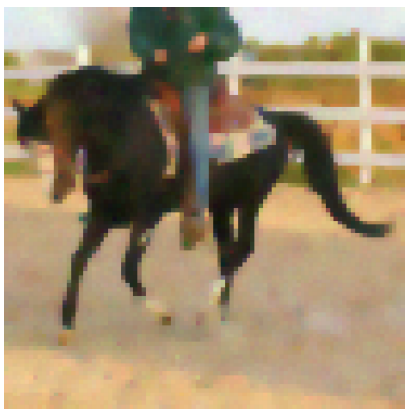




(d) Adaptive LMD (3 iterations)



(e) Adam (3 iterations)

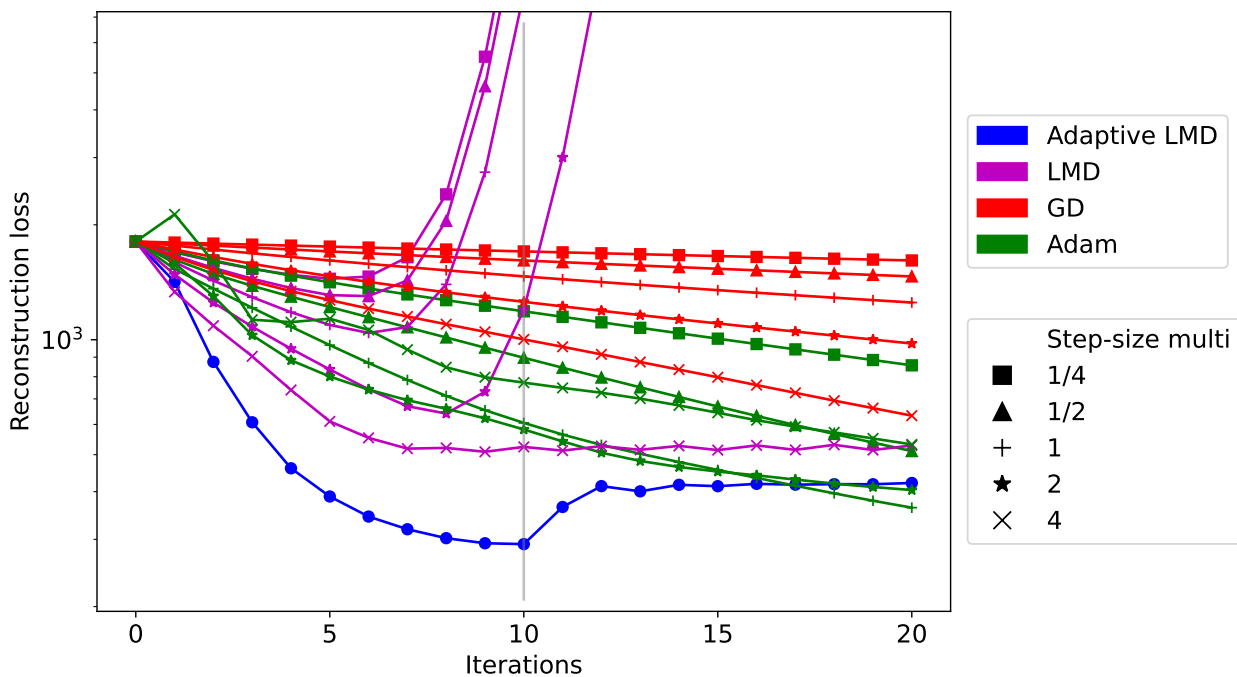


(f) Adam (10 iterations)

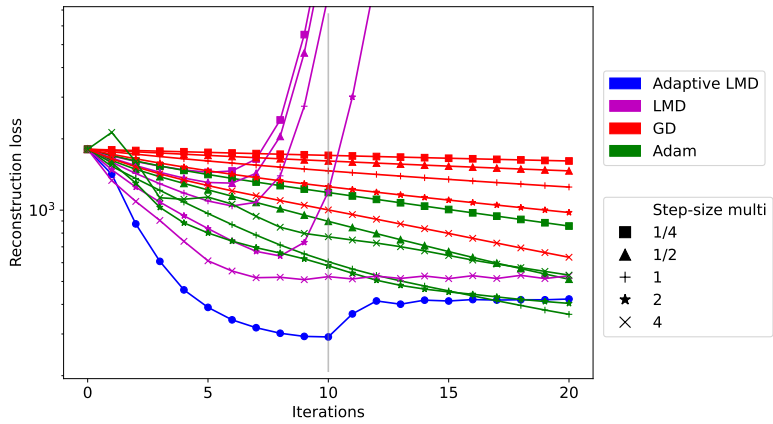
TV Inpainting

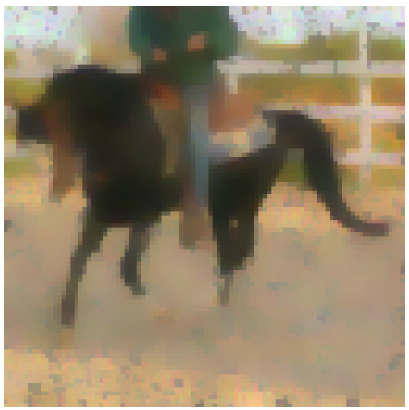
- Data: STL-10 images
- LMD trained with 20% missing pixels and noise-level $\sigma = 0.05$ for 10 iterations

$$\min_{x \in \mathcal{X}} \|Z \circ (x - y)\|_{\mathcal{X}}^2 + \lambda \|\nabla x\|_{1,\mathcal{X}}, \tag{1}$$

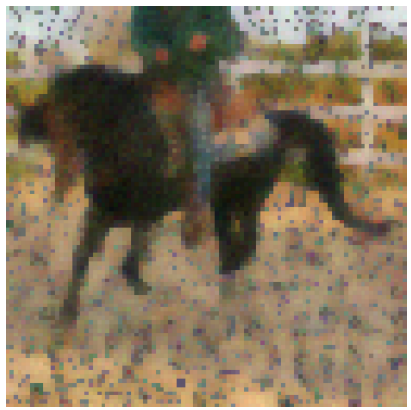


TV Inpainting





(g) Adaptive LMD (3 iterations)



(h) Adam (3 iterations)

(i) Adam (10 iterations)

Robustness

- How stable is this scheme past the learned number of iterations?
- Change of domain?
- Different forward operator?

Learned Accelerated MD

Algorithm 1: Learned Accelerated Mirror Descent (LAMD)

Data: Mirror potential Ψ, step-sizes $(t_k)_{k=1}^N > 0$, parameter $r \geq 3$
Initialize $\tilde{x}^{(0)} = x_0, \tilde{z}^{(0)} = x_0$.
for $1 \leq k \leq N$ **do**

$$\lambda_k = \frac{r}{r+k}.$$
$$x^{(k+1)} = \lambda_k \tilde{z}^{(k)} + (1-\lambda_k)\tilde{x}^{(k)}$$
$$\tilde{z}^{(k+1)} = \nabla M_\theta^*(\nabla M_\theta(\tilde{z}^{(k+1)}) - \frac{kr}{t_k}\nabla f(x^{(k+1)}))$$
$$\tilde{x}^{(k+1)} = x^{(k+1)} - \gamma t_k \nabla f(x^{(k+1)})$$

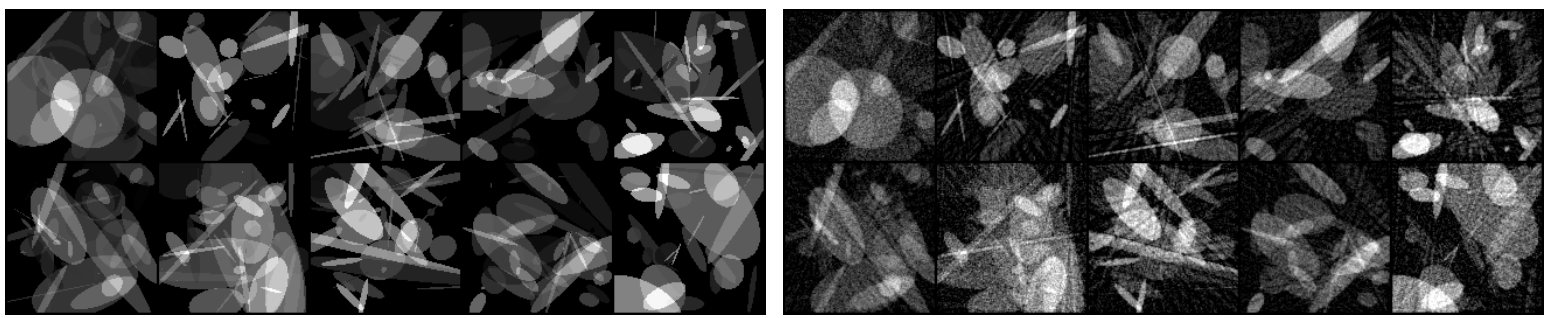
end

- Rates: AMD $\mathcal{O}(1/k^2)$, MD $\mathcal{O}(1/\sqrt{k})$

Experiment setup

- Train to denoise noisy ellipse phantoms
  - Ray transform is applied and 10% Gaussian noise added to get noisy sinograms
  - FBP is applied to noisy sinograms to get noisy ellipse phantoms
- Denoise using TV regularization

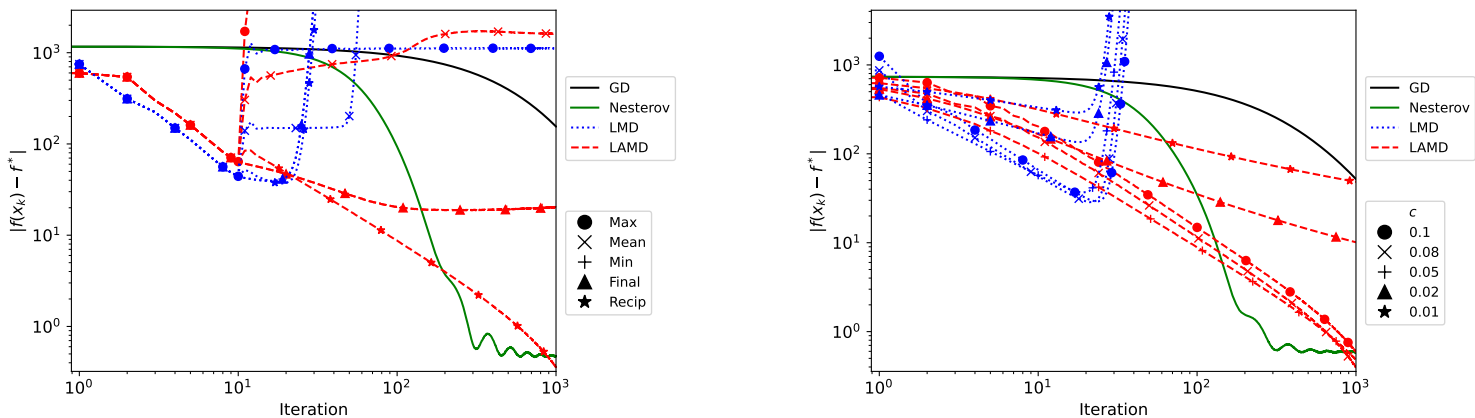
$$\min_x \|x - y\|^2 + \lambda \|\nabla x\|_1$$



Left: ground truth. Right: noisy phantoms.

LMD and LMD+momentum (LAMD) for CT

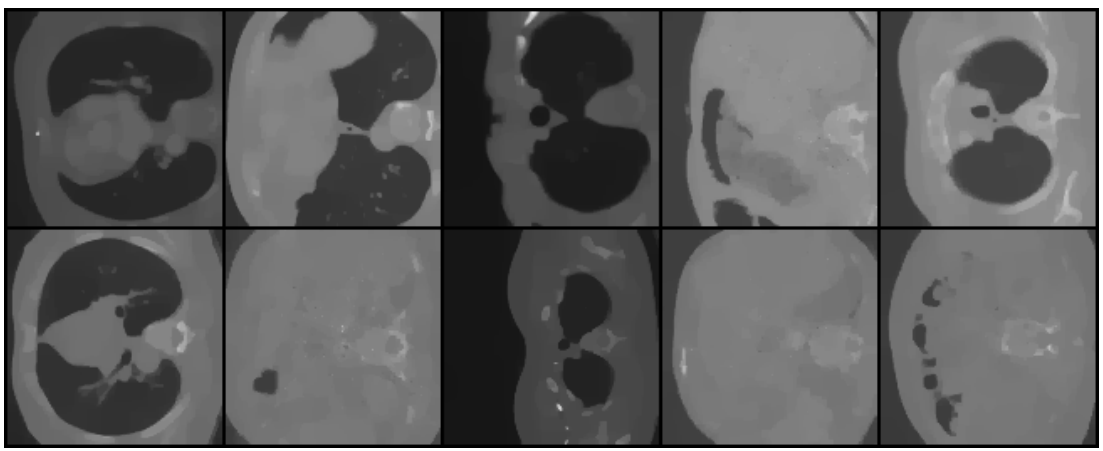
- Denoising FBP images: $\mathcal{F} = \left\{ f(x) = \frac{1}{2}\|x - y\|_2^2 + \lambda \|\nabla x\|_1 \right\}$.
- Trained LMD on ellipse phantoms, where y is FBP with parallel-beam projection.



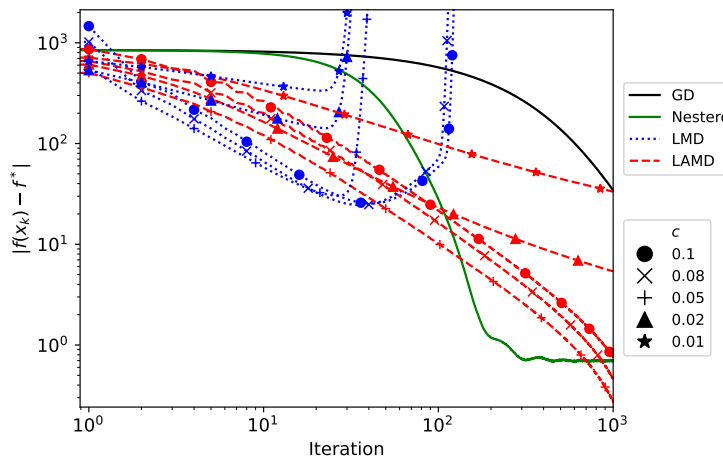
Long-term evolution of LMD and LAMD (beyond $N = 10$ that they were trained for) with various step-size extensions.

Generalizability of the learned mirror maps

- Do the learned mirror maps generalize to out-of-distribution problems? Yes, but subject to an appropriate extension of the LMD and LAMD (learned accelerated MD) iterations.



(l) LAMD on LoDoPaB Dataset

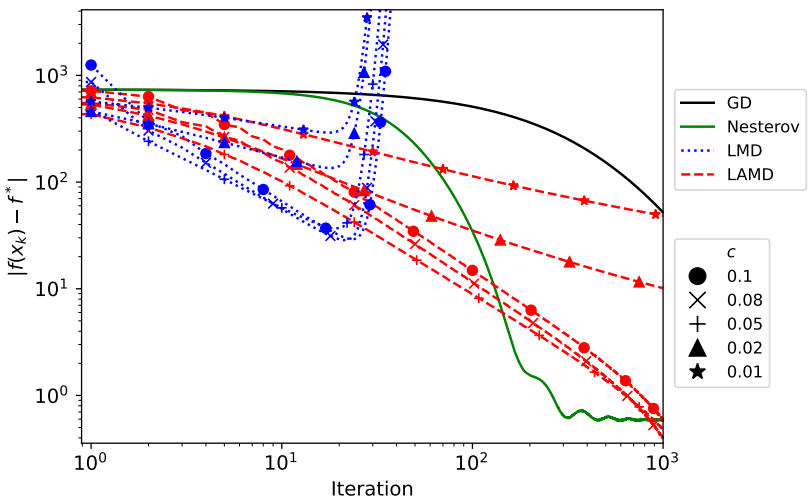


(m) Fan-beam FBP-denoising

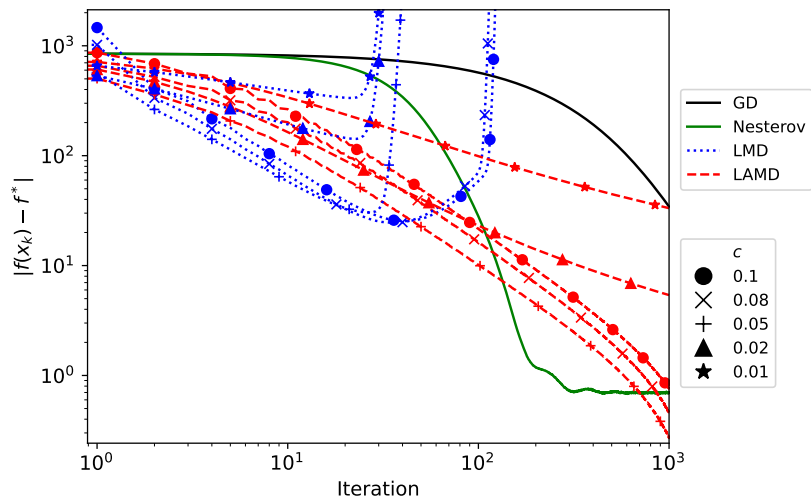
- LAMD (i.e., learned MD with momentum) has better long-term stability, and the mirror maps generalize reasonably well to a similar problem class.

Generalizability of the learned mirror maps

- Do the learned mirror maps generalize to out-of-distribution problems? Yes, but subject to an appropriate extension of the LMD and LAMD (learned accelerated MD) iterations.



(n) ellipse phantom, fan-beam FBP denoising



(o) LoDoPaB, fan-beam FBP denoising

- LAMD (i.e., learned MD with momentum) has better long-term stability, and the mirror maps generalize reasonably well to a similar problem class.

Summary and outlook

- Can use ICNNs to tailor mirror descent to the underlying geometry of the optimization manifold.
- Converges when the iterations are 'reasonably' extended beyond the training regime.
- Forward-backward inconsistency can introduce instability for later iterations.
  - Extending the iterations turns out to be stable for LMD + momentum.
  - Closed form expression for ∇M_θ^* for an ICNN M_θ?
- Extension to stochastic MD for efficiency
- Extension to non-convex problems (e.g., training a deep neural net)

Further reading

Learned Mirror Descent:

- Data-Driven Mirror Descent with Input-Convex Neural Networks. SIAM Journal on Mathematics of Data Science (SIMODS), 2023
- Robust Data-Driven Accelerated Mirror Descent. ICASSP 2023

Stochastic Deep Unrolling:

- Accelerating Deep Unrolling Networks via Dimensionality Reduction arXiv:2208.14784

Plug-and-Play Quasi-Newton:

- Provably Convergent Plug-and-Play Quasi-Newton Methods arXiv:2303.07271