A Dynamic Linesearch Method for Bilevel Learning

Joint work with Mohammad Sadegh Salehi (Bath), Matthias Ehrhardt (Bath), Subhadip Mukherjee (IIT Kharagpur)

Lindon Roberts, University of Sydney (`lindon.roberts@sydney.edu.au`)

SigmaOpt Workshop (University of South Australia)
16 February 2024

Outline

1. **Bilevel learning**
2. Dynamic linesearch
3. Numerical results

Variational Regularization

Many inverse problems can be posed in the form

$$\min_{x} \mathcal{D}(Ax, y) + \alpha \mathcal{R}(x),$$

where we wish to find x given data $y \approx Ax$.

Variational Regularization

Many inverse problems can be posed in the form

$$\min_x \mathcal{D}(Ax, y) + \alpha \mathcal{R}(x),$$

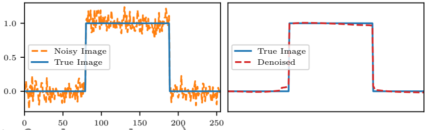
where we wish to find x given data $y \approx Ax$.

Example (image denoising): given a noisy image y, find a denoised image x by solving:

$$\min_x \underbrace{\frac{1}{2}\|x - y\|_2^2}_{\mathcal{D}(x,y)} + \alpha \underbrace{\sum_j \sqrt{\|\nabla x_j\|_2^2 + \nu^2}}_{\approx \mathrm{TV}(x)} + \frac{\xi}{2}\|x\|_2^2$$

Solution depends on choices of α, ν and ξ:

Example
($\alpha = 1$, $\nu = \xi = 10^{-3}$)



Variational Regularization

Many inverse problems can be posed in the form

$$\min_x \mathcal{D}(Ax, y) + \alpha \mathcal{R}(x),$$

where we wish to find x given data $y \approx Ax$.

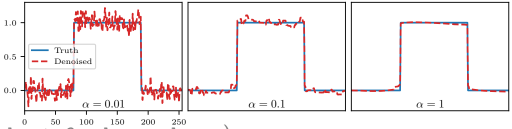
Example (image denoising): given a noisy image y, find a denoised image x by solving:

$$\min_x \underbrace{\frac{1}{2}\|x - y\|_2^2}_{\mathcal{D}(x,y)} + \alpha \underbrace{\sum_j \sqrt{\|\nabla x_j\|_2^2 + \nu^2}}_{\approx \mathrm{TV}(x)} + \frac{\xi}{2}\|x\|_2^2$$

Solution depends on choices of α, ν and ξ:

Vary α

($\nu = 10^{-3}$, $\xi = 10^{-3}$)



Variational Regularization

Many inverse problems can be posed in the form

$$\min_x \mathcal{D}(Ax, y) + \alpha \mathcal{R}(x),$$

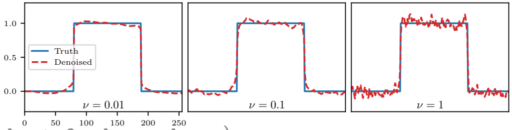
where we wish to find x given data $y \approx Ax$.

Example (image denoising): given a noisy image y, find a denoised image x by solving:

$$\min_x \underbrace{\frac{1}{2}\|x - y\|_2^2}_{\mathcal{D}(x,y)} + \alpha \underbrace{\sum_j \sqrt{\|\nabla x_j\|_2^2 + \nu^2}}_{\approx \mathrm{TV}(x)} + \frac{\xi}{2}\|x\|_2^2$$

Solution depends on choices of α, ν and ξ:

$$\frac{\text{Vary } \nu}{(\alpha = 1,\ \xi = 10^{-3})}$$



Variational Regularization

Many inverse problems can be posed in the form

$$\min_x \mathcal{D}(Ax, y) + \alpha \mathcal{R}(x),$$

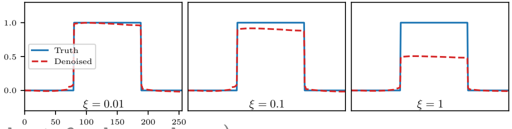
where we wish to find x given data $y \approx Ax$.

Example (image denoising): given a noisy image y, find a denoised image x by solving:

$$\min_x \underbrace{\frac{1}{2}\|x-y\|_2^2}_{\mathcal{D}(x,y)} + \alpha \underbrace{\sum_j \sqrt{\|\nabla x_j\|_2^2 + \nu^2}}_{\approx \mathrm{TV}(x)} + \frac{\xi}{2}\|x\|_2^2$$

Solution depends on choices of α, ν and ξ:

$$\frac{\text{Vary } \xi}{(\alpha = 1,\ \nu = 10^{-3})}$$



Choosing Parameters

Recovered solution depends strongly on problem parameters (e.g. α, ν and ξ)

Question

How to choose good problem parameters?

Choosing Parameters

Recovered solution depends strongly on problem parameters (e.g. α, ν and ξ)

Question

How to choose good problem parameters?

- Trial & error
- L-curve criterion
- **Bilevel learning** — data-driven approach

Suppose we have training data $(x_1, y_1), \ldots, (x_n, y_n)$ — ground truth and noisy observations.

Attempt to recover x_i from y_i by solving inverse problem with parameters $\theta \in \mathbb{R}^m$:

$$\hat{x}_i(\theta) := \arg\min_x \Phi_i(x, \theta), \qquad \text{e.g. } \Phi_i(x, \theta) = \mathcal{D}(Ax, y_i) + \theta \mathcal{R}(x).$$

Try to find θ by making $\hat{x}_i(\theta)$ close to x_i

$$\min_\theta \frac{1}{n} \sum_{i=1}^n \|\hat{x}_i(\theta) - x_i\|^2 + \mathcal{J}(\theta),$$

with optional (smooth) term $\mathcal{J}(\theta)$ to encourage particular choices of θ.

Bilevel Optimization

The bilevel learning problem is:

$$\min_{\theta} \quad f(\theta) := \frac{1}{n}\sum_{i=1}^{n} \|\hat{x}_i(\theta) - x_i\|^2 + \mathcal{J}(\theta),$$
$$\text{s.t.} \quad \hat{x}_i(\theta) := \arg\min_{x} \Phi_i(x, \theta), \quad \forall i = 1, \ldots, n.$$

- If Φ_i are strongly convex in x and sufficiently smooth in x and θ, then $\hat{x}_i(\theta)$ is well-defined and continuously differentiable.
- Upper-level problem ($\min_\theta f(\theta)$) is a smooth nonconvex optimization problem

Many use cases in data science: learning image regularizers, hyperparameter tuning, data hypercleaning, ...

Outline

1. Bilevel learning
2. **Dynamic linesearch**
3. Numerical results

Bilevel Learning

Difficulty?

Bilevel learning is just a smooth nonconvex problem — where is the challenge?

Bilevel Learning

Difficulty?

Bilevel learning is just a smooth nonconvex problem — where is the challenge?

- Can't evaluate lower-level minimizers $\hat{x}_i(\theta)$ exactly, so can never get exact $f(\theta)$ or $\nabla f(\theta)$ [Kunisch & Pock, 2013; Sherry et al., 2020]
- But can evaluate f and ∇f to arbitrary accuracy (with significant computational cost) [Berahas et al., 2021; Cao et al., 2022]
- Potentially large scale: both lower-level problems and upper-level problem.
  - Many people looking at SGD-type methods (at both levels). Not usually used for variational problems, so not a focus here. e.g. [Grazzi et al., 2021; Ji et al., 2021]

Bilevel Learning

Difficulty?

Bilevel learning is just a smooth nonconvex problem — where is the challenge?

- Can't evaluate lower-level minimizers $\hat{x}_i(\theta)$ exactly, so can never get exact $f(\theta)$ or $\nabla f(\theta)$ [Kunisch & Pock, 2013; Sherry et al., 2020]
- But can evaluate f and ∇f to arbitrary accuracy (with significant computational cost) [Berahas et al., 2021; Cao et al., 2022]
- Potentially large scale: both lower-level problems and upper-level problem.
  - Many people looking at SGD-type methods (at both levels). Not usually used for variational problems, so not a focus here. e.g. [Grazzi et al., 2021; Ji et al., 2021]

Key question: how to find good evaluation accuracy to get (i) guaranteed convergence, (ii) without requiring hyperparameter tuning, (iii) at a reasonable computational cost?

Inexact Information

First, how do we evaluate $f(\theta)$ and $\nabla f(\theta)$? [Ehrhardt & LR, 2023]

- $\hat{x}(\theta)$ is minimiser of smooth, strongly convex problem — given ϵ, use standard first-order methods (e.g. GD) to get $x_\epsilon = x_\epsilon(\theta)$ with $\|x_\epsilon - \hat{x}(\theta)\| \leq \epsilon$

Inexact Information

First, how do we evaluate $f(\theta)$ and $\nabla f(\theta)$? [Ehrhardt & LR, 2023]

- $\hat{x}(\theta)$ is minimiser of smooth, strongly convex problem — given ϵ, use standard first-order methods (e.g. GD) to get $x_\epsilon = x_\epsilon(\theta)$ with $\|x_\epsilon - \hat{x}(\theta)\| \leq \epsilon$
- For an objective $g(\hat{x}(\theta))$, Implicit Function Theorem gives

$$\nabla_\theta g = -[\partial_x \partial_\theta \Phi(\hat{x}(\theta), \theta)]^T [\partial_{xx} \Phi(\hat{x}(\theta), \theta)]^{-1} \nabla_x g(\hat{x}(\theta))$$

Inexact Information

First, how do we evaluate $f(\theta)$ and $\nabla f(\theta)$? [Ehrhardt & LR, 2023]

- $\hat{x}(\theta)$ is minimiser of smooth, strongly convex problem — given ϵ, use standard first-order methods (e.g. GD) to get $x_\epsilon = x_\epsilon(\theta)$ with $\|x_\epsilon - \hat{x}(\theta)\| \leq \epsilon$
- For an objective $g(\hat{x}(\theta))$, Implicit Function Theorem gives

$$\nabla_\theta g = -[\partial_x \partial_\theta \Phi(\hat{x}(\theta), \theta)]^T [\partial_{xx} \Phi(\hat{x}(\theta), \theta)]^{-1} \nabla_x g(\hat{x}(\theta))$$

- Given δ, use CG to find $q_{\epsilon,\delta}$ such that $\|[\partial_{xx}\Phi(x_\epsilon, \theta)] q_{\epsilon,\delta} - \nabla_x g(x_\epsilon)\| \leq \delta$
- Use approximate gradient $-[\partial_x \partial_\theta \Phi(x_\epsilon, \theta)]^T q_{\epsilon,\delta}$

Inexact Information

First, how do we evaluate $f(\theta)$ and $\nabla f(\theta)$? [Ehrhardt & LR, 2023]

- $\hat{x}(\theta)$ is minimiser of smooth, strongly convex problem — given ϵ, use standard first-order methods (e.g. GD) to get $x_\epsilon = x_\epsilon(\theta)$ with $\|x_\epsilon - \hat{x}(\theta)\| \leq \epsilon$
- For an objective $g(\hat{x}(\theta))$, Implicit Function Theorem gives

$$\nabla_\theta g = -[\partial_x \partial_\theta \Phi(\hat{x}(\theta), \theta)]^T [\partial_{xx} \Phi(\hat{x}(\theta), \theta)]^{-1} \nabla_x g(\hat{x}(\theta))$$

- Given δ, use CG to find $q_{\epsilon,\delta}$ such that $\|[\partial_{xx}\Phi(x_\epsilon, \theta)] q_{\epsilon,\delta} - \nabla_x g(x_\epsilon)\| \leq \delta$
- Use approximate gradient $-[\partial_x \partial_\theta \Phi(x_\epsilon, \theta)]^T q_{\epsilon,\delta}$
- Total gradient error is $\mathcal{O}(\epsilon + \delta + \epsilon^2 + \epsilon\delta)$ with computable constants

Note: this is equivalent to an accelerated version of backpropagation applied to the lower-level solver iteration. [Mehmood & Ochs, 2020]

Linesearch Framework

The underlying algorithmic approach is gradient descent with backtracking Armijo linesearch: e.g. [Nocedal & Wright, 2006]

For $j = 0, 1, 2, \ldots,$

- New candidate point $\hat{\theta} = \theta_k - \alpha\rho^j \nabla f(\theta_k)$, some $\alpha > 0$ and $\rho \in (0, 1)$.
- Check for sufficient decrease:

$$f(\hat{\theta}) \leq f(\theta_k) - \lambda\alpha\rho^j \|\nabla f(\theta_k)\|^2,$$

  for some $\lambda \in (0, 1)$.
- If sufficient decrease, $\theta_{k+1} = \hat{\theta}$ and stop loop; otherwise, try next value of j.

Linesearch Framework

The underlying algorithmic approach is gradient descent with backtracking Armijo linesearch: e.g. [Nocedal & Wright, 2006]

For $j = 0, 1, 2, \ldots,$

- New candidate point $\hat{\theta} = \theta_k - \alpha\rho^j \nabla f(\theta_k)$, some $\alpha > 0$ and $\rho \in (0, 1)$.
- Check for sufficient decrease:

$$f(\hat{\theta}) \leq f(\theta_k) - \lambda\alpha\rho^j \|\nabla f(\theta_k)\|^2,$$

  for some $\lambda \in (0, 1)$.
- If sufficient decrease, $\theta_{k+1} = \hat{\theta}$ and stop loop; otherwise, try next value of j.

Basic proof ideas: Taylor's theorem and $\lambda < 1$ guarantee some j eventually gives sufficient decrease. Slow decrease in stepsize $\alpha\rho^j$ guarantees stepsize never too small, so $f(\theta_k) - f(\theta_{k+1}) \geq \mathcal{O}(\|\nabla f(\theta_k)\|^2)$.

Inexact Linesearch

To handle inexactness, there are two key issues to resolve:

- Given $z_k \approx \nabla f(\theta_k)$ can we guarantee z_k is a descent direction?
- If no sufficient decrease (with inexact $f(\theta)$ evaluations), should we shrink stepsize or improve accuracy in f (or ∇f)?

Inexact Linesearch

To handle inexactness, there are two key issues to resolve:

- Given $z_k \approx \nabla f(\theta_k)$ can we guarantee z_k is a descent direction?
- If no sufficient decrease (with inexact $f(\theta)$ evaluations), should we shrink stepsize or improve accuracy in f (or ∇f)?

To be practical, we don't want to make accuracy in f or ∇f unnecessarily high (but don't want to lose convergence guarantees either).

Inexact Gradient Calculation

- Given ϵ and δ, calculate inexact lower-level minimiser x_ϵ and inexact gradient $z_k \approx \nabla f(\theta_k)$ (using CG with residual tolerance δ)
- Calculate computable upper bound ω for $\|z_k - \nabla f(\theta_k)\|$
- If $\omega \leq (1-\eta)\|z_k\|$, then use z_k (guaranteed descent direction)
- Otherwise, decrease ϵ and δ by a constant factor and start again

Inexact Gradient Calculation

- Given ϵ and δ, calculate inexact lower-level minimiser x_ϵ and inexact gradient $z_k \approx \nabla f(\theta_k)$ (using CG with residual tolerance δ)
- Calculate computable upper bound ω for $\|z_k - \nabla f(\theta_k)\|$
- If $\omega \le (1-\eta)\|z_k\|$, then use z_k (guaranteed descent direction)
- Otherwise, decrease ϵ and δ by a constant factor and start again

Theorem

If $\|\nabla f(\theta_k)\| \neq 0$, then z_k is a descent direction for all sufficiently small ϵ and δ.

i.e. Gradient calculation terminates in finite time.

Sufficient Decrease Condition

Inexact sufficient decrease condition

- Given $\hat{\theta} = \theta_k - \alpha_k z_k$, compute $x_\epsilon(\theta_k)$ and $x_\epsilon(\hat{\theta})$ to accuracy ϵ
- Compute approximate objective values $\tilde{f}(\theta_k)$ and $\tilde{f}(\hat{\theta})$
- Inexact sufficient decrease condition is (for L-smooth and convex f):

$$\tilde{f}(\hat{\theta}) \leq \tilde{f}(\theta_k) - \lambda\alpha_k\|z_k\|^2 - \|\nabla_x f(x_\epsilon(\hat{\theta}))\|\epsilon - \|\nabla_x f(x_\epsilon(\theta_k))\|\epsilon - \frac{1}{2}L\epsilon^2$$

Sufficient Decrease Condition

Inexact sufficient decrease condition

- Given $\hat{\theta} = \theta_k - \alpha_k z_k$, compute $x_\epsilon(\theta_k)$ and $x_\epsilon(\hat{\theta})$ to accuracy ϵ
- Compute approximate objective values $\tilde{f}(\theta_k)$ and $\tilde{f}(\hat{\theta})$
- Inexact sufficient decrease condition is (for L-smooth and convex f):

$$\tilde{f}(\hat{\theta}) \leq \tilde{f}(\theta_k) - \lambda \alpha_k \|z_k\|^2 - \|\nabla_x f(x_\epsilon(\hat{\theta}))\|\epsilon - \|\nabla_x f(x_\epsilon(\theta_k))\|\epsilon - \frac{1}{2} L \epsilon^2$$

Theorem

- *If inexact sufficient decrease condition holds, then* $f(\hat{\theta}) \leq f(\theta_k) - \lambda \alpha_k \|z_k\|^2$.

Sufficient Decrease Condition

Inexact sufficient decrease condition

- Given $\hat{\theta} = \theta_k - \alpha_k z_k$, compute $x_\epsilon(\theta_k)$ and $x_\epsilon(\hat{\theta})$ to accuracy ϵ
- Compute approximate objective values $\tilde{f}(\theta_k)$ and $\tilde{f}(\hat{\theta})$
- Inexact sufficient decrease condition is (for L-smooth and convex f):

$$\tilde{f}(\hat{\theta}) \leq \tilde{f}(\theta_k) - \lambda \alpha_k \|z_k\|^2 - \|\nabla_x f(x_\epsilon(\hat{\theta}))\|\epsilon - \|\nabla_x f(x_\epsilon(\theta_k))\|\epsilon - \frac{1}{2}L\epsilon^2$$

Theorem

- *If inexact sufficient decrease condition holds, then* $f(\hat{\theta}) \leq f(\theta_k) - \lambda \alpha_k \|z_k\|^2$.
- *For any* ϵ, *inexact sufficient decrease condition holds for all* $\alpha_k \in [\alpha_{\min}(\epsilon), \alpha_{\max}(\epsilon)]$

Sufficient Decrease Condition

Inexact sufficient decrease condition

- Given $\hat{\theta} = \theta_k - \alpha_k z_k$, compute $x_\epsilon(\theta_k)$ and $x_\epsilon(\hat{\theta})$ to accuracy ϵ
- Compute approximate objective values $\tilde{f}(\theta_k)$ and $\tilde{f}(\hat{\theta})$
- Inexact sufficient decrease condition is (for L-smooth and convex f):

$$\tilde{f}(\hat{\theta}) \leq \tilde{f}(\theta_k) - \lambda\alpha_k\|z_k\|^2 - \|\nabla_x f(x_\epsilon(\hat{\theta}))\|\epsilon - \|\nabla_x f(x_\epsilon(\theta_k))\|\epsilon - \frac{1}{2}L\epsilon^2$$

Theorem

- *If inexact sufficient decrease condition holds, then* $f(\hat{\theta}) \leq f(\theta_k) - \lambda\alpha_k\|z_k\|^2$.
- *For any* ϵ, *inexact sufficient decrease condition holds for all* $\alpha_k \in [\alpha_{\min}(\epsilon), \alpha_{\max}(\epsilon)]$
- *As* $\epsilon \to 0$, *we have* $[\alpha_{\min}(\epsilon), \alpha_{\max}(\epsilon)] \to [0, \alpha_{\max}]$ *for some* $\alpha_{\max} > 0$

Inexact Backtracking (single iteration k)

1: **for** $J_{\max} = J_0, J_0+1, J_0+2, \ldots$ **do**
2: Compute inexact gradient z_k (possibly reducing ϵ and δ)
3: **for** $j = 0, \ldots, J_{\max}-1$ **do**
4: If sufficient decrease with stepsize $\alpha_k = \alpha\rho^j$, go to line 8
5: **end for**
6: Reduce ϵ and δ by constant factor *(backtracking failed, need higher accuracy)*
7: **end for**
8: Set $\theta_{k+1} = \theta_k - \alpha_k z_k$ *(successful linesearch)*
9: Increase ϵ and δ by constant factor for next iteration

Inexact Backtracking

Inexact Backtracking (single iteration k)

1: **for** $J_{\text{max}} = J_0, J_0 + 1, J_0 + 2, \ldots$ **do**
2: Compute inexact gradient z_k (possibly reducing ϵ and δ)
3: **for** $j = 0, \ldots, J_{\text{max}} - 1$ **do**
4: If sufficient decrease with stepsize $\alpha_k = \alpha\rho^j$, go to line 8
5: **end for**
6: Reduce ϵ and δ by constant factor *(backtracking failed, need higher accuracy)*
7: **end for**
8: Set $\theta_{k+1} = \theta_k - \alpha_k z_k$ *(successful linesearch)*
9: Increase ϵ and δ by constant factor for next iteration

Theorem

At each iteration k, successful linesearch occurs in finite time. Hence $\|\nabla f(\theta_k)\| \to 0$.

Outline

1. Bilevel learning
2. Dynamic linesearch
3. **Numerical results**

Quadratic Problem

Simple linear least-squares problem (closed form for true solution):

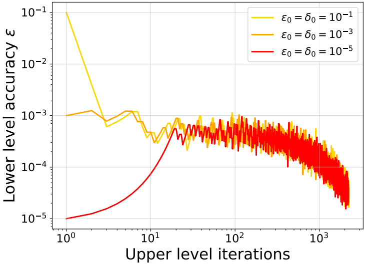
$$\min_{\theta} \; f(\theta) := \|A_1 \hat{x}(\theta) - b_1\|^2 \qquad \text{s.t. } \hat{x}(\theta) = \arg\min_{x} \Phi(x, \theta) := \|A_2 x + A_3 \theta - b_2\|^2$$

Quadratic Problem

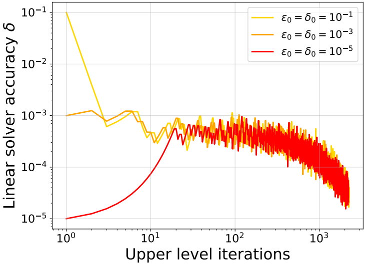
Simple linear least-squares problem (closed form for true solution):

$$\min_{\theta} \; f(\theta) := \|A_1 \hat{x}(\theta) - b_1\|^2 \qquad \text{s.t. } \hat{x}(\theta) = \arg\min_x \Phi(x, \theta) := \|A_2 x + A_3 \theta - b_2\|^2$$

Do hyperparameters (initial accuracies ϵ and δ) matter?



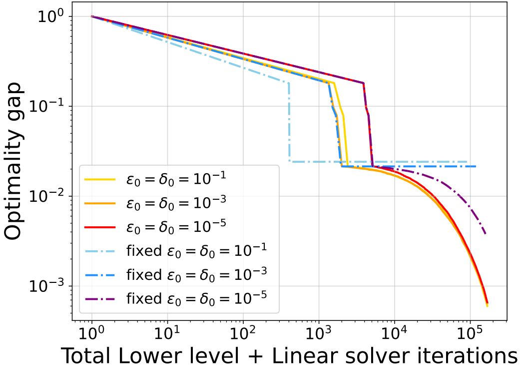
Final ϵ at each iteration



Final δ at each iteration

Quadratic Problem

Dynamic accuracy is better than fixed accuracy



Optimality gap vs. computational work (lower-level + CG iterations)

Field of Experts Image Denoising

$$\min_{\theta} \; f(\theta) := \frac{1}{N}\sum_{i=1}^{N} \|\hat{x}_i(\theta) - x_i^*\|^2,$$

$$\text{s.t. } \hat{x}_i(\theta) = \arg\min_{x} \Phi_i(x,\theta) := \frac{1}{2}\|x - y_i\|^2 + \sum_{k=1}^{K} \beta_k(\theta)\|c_k(\theta) * x\|_{k,\theta} + \frac{\mu}{2}\|x\|^2.$$

Learn $K = 30$ filters $c_k(\theta)$, smoothed ℓ_1-norms $\|\cdot\|_{k,\theta}$ and weights $\beta_k(\theta)$ to reconstruct noisy 2D images (≈ 1500 hyperparameters θ).

Using $N = 25$ training images (x_i^*, y_i) of size 96×96 pixels.

Field of Experts Denoising

Apply learned filters on new test image

True image

Noisy (PSNR 20.0dB)

Denoised (PSNR 28.7dB)

(Palladian Bridge, Bath, UK)

Conclusions & Future Work

Conclusions

- Bilevel learning provides a structured hyperparameter tuning method
- New linesearch method balances accuracy and computational efficiency
- Strong practical performance and robust to algorithm parameter choices
  - Outperforms other existing approaches (e.g. prescribed accuracy schedule, inexact derivative-free methods) [Pedregosa, 2016; Ehrhardt & LR, 2021]

Future Work

- Handle large training sets with SGD-type methods
- Extensions to non-strongly convex lower-level problems

Preprint: `https://arxiv.org/abs/2308.10098` (substantial revisions coming soon)

References i

A. S. Berahas, L. Cao, and K. Scheinberg, *Global convergence rate analysis of a generic line search algorithm with noise*, SIAM Journal on Optimization, 31 (2021), pp. 1489–1518.

L. Cao, A. S. Berahas, and K. Scheinberg, *First- and second-order high probability complexity bounds for trust-region methods with noisy oracles*, arXiv preprint 2205.03667, (2022).

M. J. Ehrhardt and L. Roberts, *Inexact derivative-free optimization for bilevel learning*, Journal of Mathematical Imaging and Vision, 63 (2021), pp. 580–600.

M. J. Ehrhardt and L. Roberts, *Analyzing inexact hypergradients for bilevel learning*, IMA Journal of Applied Mathematics, (2023).

R. Grazzi, M. Pontil, and S. Salzo, *Convergence properties of stochastic hypergradients*, in Proceedings of the 24th International Conference on Artificial Intelligence and Statistics (AISTATS) 2021, vol. 130, 2021, pp. 3826–3834.

K. Ji, J. Yang, and Y. Liang, *Bilevel optimization for machine learning: Algorithm design and convergence analysis*, in Proceedings of the 38th International Conference on Machine Learning, 2021, pp. 4882–4892.

K. Kunisch and T. Pock, *A Bilevel Optimization Approach for Parameter Learning in Variational Models*, SIAM Journal on Imaging Sciences, 6 (2013), pp. 938–983.

S. Mehmood and P. Ochs, *Automatic differentiation of some first-order methods in parametric optimization*, in Proceedings of the 23rd International Conference on Artificial Intelligence and Statistics (AISTATS), Palermo, Italy, 2020.

J. Nocedal and S. J. Wright, *Numerical Optimization*, Springer Series in Operations Research and Financial Engineering, Springer, New York, 2nd ed., 2006.

F. Pedregosa, *Hyperparameter optimization with approximate gradient*, in Proceedings of the 33rd International Conference on Machine Learning, New York, 2016.

F. Sherry, M. Benning, J. C. De los Reyes, M. J. Graves, G. Maierhofer, G. Williams, C.-B. Schonlieb, and M. J. Ehrhardt, *Learning the sampling pattern for MRI*, IEEE Transactions on Medical Imaging, 39 (2020), pp. 4310–4321.