Enhanced Bilevel Optimization via Bregman Distance

Feihu Huang[1,2], Junyi Li[1], Shangqian Gao[1], Heng Huang[1]

1. University of Pittsburgh
2. Nanjing University of Aeronautics & Astronautics

NeurIPS | 2022

Thirty-sixth Conference on Neural Information Processing Systems

NEURAL INFORMATION PROCESSING SYSTEMS

Roadmap

- Background
- Enhanced Bilevel Optimization via Bregman Distance
- Convergence Properties
- Experimental Results
- Conclusions

Roadmap

- Background
- Enhanced Bilevel Optimization via Bregman Distance
- Convergence Properties
- Experimental Results
- Conclusions

Background

- Bilevel optimization can effectively solve the problems with a hierarchical structure.
- So it recently has been widely used in many machine learning tasks such as hyper-parameter optimization, meta learning and reinforcement learning.

$$\min_{x\in\mathcal{X}\subseteq\mathbb{R}^{d_1}} f(x, y^*(x)) + h(x),$$
$$\text{s.t. } y^*(x) \in \arg\min_{y\in\mathbb{R}^{d_2}} g(x, y),$$

Background

- For example, the hyper-representation learning could be seen as a generalized meta learning, defined as:

$$\min_{\lambda} l_{val}(\lambda, w^*(\lambda)) := \mathbb{E}_{\xi}\left[\frac{1}{|D_{\mathcal{V},\xi}|}\sum_{(x_i,y_i)\in D_{\mathcal{V},\xi}} l\Big(w^*_{\xi}(\lambda)^T\phi(x_i;\lambda), y_i\Big);\xi\right] + \alpha\|\lambda\|_1$$

$$\text{s.t. } w^*_{\xi}(\lambda) = \arg\min_{w} l_{tr}(\lambda, w;\xi) := \frac{1}{|D_{\mathcal{T},\xi}|}\sum_{(x_i,y_i)\in D_{\mathcal{T},\xi}} l\Big(w^T\phi(x_i;\lambda), y_i\Big) + C\|w\|^2,$$

where $l(\cdot)$ denotes the cross entropy loss, $D_{\mathcal{T},\xi}$ and $D_{\mathcal{V},\xi}$ are training and validation datasets for randomly sampled meta task ξ. Here $\phi(\cdot,\cdot)$ is a four-layers convolutional neural network with max-pooling and 32 filters per layer [9], which denotes a representation mapping. λ denotes the parameter

Roadmap

- Background
- Enhanced Bilevel Optimization via Bregman Distance
- Convergence Properties
- Experimental Results
- Conclusions

Bregman Distance

Given a ρ-strongly convex and continuously-differentiable function $\psi(x)$, *i.e.*, $\langle x_1 - x_2, \nabla\psi(x_1) - \nabla\psi(x_2)\rangle \geq \rho\|x_1 - x_2\|^2$, we define a Bregman distance [3, 4] for any $x_1, x_2 \in \mathcal{X}$:

$$D_\psi(x_1, x_2) = \psi(x_1) - \psi(x_2) - \langle \nabla\psi(x_2), x_1 - x_2\rangle.$$

In Algorithm 1, we use the mirror descent iteration to update the variable x at $t+1$-th step:

$$x_{t+1} = \arg\min_{x\in\mathcal{X}} \Big\{\langle w_t, x\rangle + h(x) + \frac{1}{\gamma} D_{\psi_t}(x, x_t)\Big\}, \tag{5}$$

where $\gamma > 0$ is stepsize, and w_t is an estimator of $\nabla F(x_t)$. Here the mirror function ψ_t can be dynamic as the algorithm is running. Let $\psi_t(x) = \frac{1}{2}\|x\|^2$, we have $D_{\psi_t}(x, x_t) = \frac{1}{2}\|x - x_t\|^2$. When $\mathcal{X} = \mathbb{R}^{d_1}$, the above subproblem (5) is equivalent to the proximal gradient descent. When $\mathcal{X} \subseteq \mathbb{R}^{d_1}$ and $h(x) = 0$, the above subproblem (5) is equivalent to the projection gradient descent.

Enhanced Bilevel Optimization via Bregman Distance

$$\min_{x \in \mathcal{X} \subseteq \mathbb{R}^{d_1}} \quad f(x, y^*(x)) + h(x),$$

$$\text{s.t. } y^*(x) \in \arg \min_{y \in \mathbb{R}^{d_2}} \; g(x, y),$$

Algorithm 1 Deterministic BiO-BreD Algorithm

Input: T, $K \geq 1$, learning rates $\gamma > 0$, $\lambda > 0$;
initialize: $x_0 \in \mathcal{X}$ and $y_{-1}^K = y_0 \in \mathbb{R}^{d_2}$;
for $t = 0, 1, \cdots, T-1$ **do**
    Let $y_t^0 = y_{t-1}^K$;
    for $k = 1, \cdots, K$ **do**
        Update $y_t^k = y_t^{k-1} - \lambda \nabla_y g(x_t, y_t^{k-1})$;
    end for
    Compute partial derivative $w_t = \frac{\partial f(x_t, y_t^K)}{\partial x}$ via backpropagation *w.r.t.* x_t;
    Given a ρ-strongly convex mirror function ψ_t;
    Update $\boxed{x_{t+1} = \arg\min_{x \in \mathcal{X}} \left\{ \langle w_t, x \rangle + h(x) + \frac{1}{\gamma} D_{\psi_t}(x, x_t) \right\};}$
end for
Output: Uniformly and randomly choose from $\{x_t, y_t\}_{t=1}^T$.

Enhanced Bilevel Optimization via Bregman Distance

$$\min_{x\in\mathcal{X}\subseteq\mathbb{R}^{d_1}} \mathbb{E}_{\xi\sim\mathcal{D}}\big[f(x, y^*(x);\xi)\big] + h(x),$$

$$\text{s.t. } y^*(x) \in \arg\min_{y\in\mathbb{R}^{d_2}} \mathbb{E}_{\zeta\sim\mathcal{D}'}\big[g(x, y;\zeta)\big],$$

Algorithm 2 Stochastic BiO-BreD (SBiO-BreD) Algorithm

Input: $T, K \geq 1$, stepsizes $\gamma > 0$, $\lambda > 0$, $\{\eta_t\}_{t=1}^T$;
initialize: $x_0 \in \mathcal{X}$ and $y_0 \in \mathbb{R}^{d_2}$;
for $t = 0, 1, \cdots, T-1$ **do**
  Draw randomly b independent samples $\mathcal{B}_t = \{\zeta_t^i\}_{i=1}^b$, and compute stochastic partial derivatives $v_t = \nabla_y g(x_t, y_t; \mathcal{B}_t)$;
  Update $y_{t+1} = y_t - \lambda\eta_t v_t$;
  Draw randomly $b(K+1)$ independent samples $\bar{\mathcal{B}}_t = \{\xi_{t,i}, \zeta_{t,i}^0 \cdots, \zeta_{t,i}^{K-1}\}_{i=1}^b$, and compute stochastic partial derivatives $w_t = \bar{\nabla} f(x_t, y_t; \bar{\mathcal{B}}_t)$;
  Given a ρ-strongly convex mirror function ψ_t;
  Update $\boxed{x_{t+1} = \arg\min_{x\in\mathcal{X}} \big\{\langle w_t, x\rangle + h(x) + \frac{1}{\gamma}D_{\psi_t}(x, x_t)\big\};}$
end for
Output: Uniformly and randomly choose from $\{x_t, y_t\}_{t=1}^T$.

Enhanced Bilevel Optimization via Bregman Distance

Algorithm 3 Accelerated Stochastic BiO-BreD (ASBiO-BreD) Algorithm

Input: $T, K \geq 1$, q, stepsizes $\gamma > 0$, $\lambda > 0$, $\{\eta_t\}_{t=1}^T$, mini-batch sizes b and b_1;
initialize: $x_0 \in \mathcal{X}$ and $y_0 \in \mathbb{R}^{d_2}$;
for $t = 0, 1, \cdots, T-1$ **do**
  if $\mod (t, q) = 0$ **then**
    Draw randomly b independent samples $\mathcal{B}_t = \{\zeta_t^i\}_{i=1}^b$, and compute stochastic partial derivative $v_t = \nabla_y g(x_t, y_t; \mathcal{B}_t)$;
    Draw randomly $b(K+1)$ independent samples $\bar{\mathcal{B}}_t = \{\xi_{t,i}, \zeta_{t,i}^0 \cdots, \zeta_{t,i}^{K-1}\}_{i=1}^b$, and compute stochastic partial derivative $w_t = \bar{\nabla} f(x_t, y_t; \bar{\mathcal{B}}_t)$;
  else
    Generate randomly b_1 independent samples $\mathcal{I}_t = \{\zeta_t^i\}_{i=1}^{b_1}$, and compute stochastic partial derivative $v_t = \nabla_y g(x_t, y_t; \mathcal{I}_t) - \nabla_y g(x_{t-1}, y_{t-1}; \mathcal{I}_t) + v_{t-1}$;
    Generate randomly $b_1(K+1)$ independent samples $\bar{\mathcal{I}}_t = \{\xi_{t,i}, \zeta_{t,i}^0 \cdots, \zeta_{t,i}^{K-1}\}_{i=1}^{b_1}$, and compute stochastic partial derivative $w_t = \bar{\nabla} f(x_t, y_t; \bar{\mathcal{I}}_t) - \bar{\nabla} f(x_{t-1}, y_{t-1}; \bar{\mathcal{I}}_t) + w_{t-1}$;
  end if
  Update $y_{t+1} = y_t - \lambda \eta_t v_t$;
  Given a ρ-strongly convex mirror function ψ_t;
  Update $\boxed{x_{t+1} = \arg\min_{x \in \mathcal{X}} \left\{ \langle w_t, x \rangle + h(x) + \frac{1}{\gamma} D_{\psi_t}(x, x_t) \right\};}$
end for
Output: Uniformly and randomly choose from $\{x_t, y_t\}_{t=1}^T$.

Roadmap

- Background
- Enhanced Bilevel Optimization via Bregman Distance
- Convergence Properties
- Experimental Results
- Conclusions

Convergence Properties

Table 1: Comparisons of the representative bilevel optimization algorithms for finding an ϵ-stationary point of the **deterministic** nonconvex-strongly-convex Problem (1) with $h(x)$ or without $h(x)$, *i.e.*, $\|\nabla F(x)\|^2 \leq \epsilon$ or its equivalent variants. $Gc(f,\epsilon)$ and $Gc(g,\epsilon)$ denote the number of gradient evaluations *w.r.t.* $f(x,y)$ and $g(x,y)$; $JV(g,\epsilon)$ denotes the number of Jacobian-vector products; $HV(g,\epsilon)$ is the number of Hessian-vector products; $\kappa = L/\mu$ is the conditional number. $\surd$ means that the algorithms solve both the **smooth** and **nonsmooth** bilevel optimizations.

Algorithm	**Reference**	$Gc(f,\epsilon)$	$Gc(g,\epsilon)$	$JV(g,\epsilon)$	$HV(g,\epsilon)$	**Nonsmooth**
AID-BiO	[11]	$O(\kappa^4\epsilon^{-1})$	$O(\kappa^5\epsilon^{-5/4})$	$O(\kappa^4\epsilon^{-1})$	$O(\kappa^{4.5}\epsilon^{-1})$	
AID-BiO	[22]	$O(\kappa^3\epsilon^{-1})$	$O(\kappa^4\epsilon^{-1})$	$O(\kappa^3\epsilon^{-1})$	$O(\kappa^{3.5}\epsilon^{-1})$	
ITD-BiO	[22]	$O(\kappa^3\epsilon^{-1})$	$\tilde{O}(\kappa^4\epsilon^{-1})$	$\tilde{O}(\kappa^4\epsilon^{-1})$	$\tilde{O}(\kappa^4\epsilon^{-1})$	
BiO-BreD	Ours	$O(\kappa^2\epsilon^{-1})$	$\tilde{O}(\kappa^3\epsilon^{-1})$	$\tilde{O}(\kappa^3\epsilon^{-1})$	$\tilde{O}(\kappa^3\epsilon^{-1})$	$\surd$

$$\min_{x\in\mathcal{X}\subseteq\mathbb{R}^{d_1}} f(x, y^*(x)) + h(x),$$

$$\text{s.t. } y^*(x) \in \arg\min_{y\in\mathbb{R}^{d_2}} g(x,y),$$

Convergence Properties

Table 2: Comparisons of the representative bilevel optimization algorithms for finding an ϵ-stationary point of the **stochastic** nonconvex-strongly-convex problem (2) with $h(x)$ or without $h(x)$, *i.e.*, $\mathbb{E}\|\nabla F(x)\|^2 \leq \epsilon$ or its equivalent variants. Since some algorithms do not provide the explicit dependence on κ, we use $p(\kappa)$.

Algorithm	**Reference**	$Gc(f,\epsilon)$	$Gc(g,\epsilon)$	$JV(g,\epsilon)$	$HV(g,\epsilon)$	**Nonsmooth**
TTSA	[15]	$O(p(\kappa)\epsilon^{-2.5})$	$O(p(\kappa)\epsilon^{-2.5})$	$O(p(\kappa)\epsilon^{-2.5})$	$O(p(\kappa)\epsilon^{-2.5})$	
STABLE	[5]	$O(p(\kappa)\epsilon^{-2})$	$O(p(\kappa)\epsilon^{-2})$	$O(p(\kappa)\epsilon^{-2})$	$O(p(\kappa)\epsilon^{-2})$	
SMB	[13]	$O(p(\kappa)\epsilon^{-2})$	$O(p(\kappa)\epsilon^{-2})$	$O(p(\kappa)\epsilon^{-2})$	$O(p(\kappa)\epsilon^{-2})$	
VRBO	[41]	$O(p(\kappa)\epsilon^{-1.5})$	$O(p(\kappa)\epsilon^{-1.5})$	$O(p(\kappa)\epsilon^{-1.5})$	$O(p(\kappa)\epsilon^{-1.5})$	
SUSTAIN	[23]	$O(p(\kappa)\epsilon^{-1.5})$	$O(p(\kappa)\epsilon^{-1.5})$	$O(p(\kappa)\epsilon^{-1.5})$	$O(p(\kappa)\epsilon^{-1.5})$	
VR-saBiAdam	[18]	$O(p(\kappa)\epsilon^{-1.5})$	$O(p(\kappa)\epsilon^{-1.5})$	$O(p(\kappa)\epsilon^{-1.5})$	$O(p(\kappa)\epsilon^{-1.5})$	
BSA	[11]	$O(\kappa^6\epsilon^{-2})$	$O(\kappa^9\epsilon^{-3})$	$O(\kappa^6\epsilon^{-2})$	$\tilde{O}(\kappa^6\epsilon^{-2})$	
stocBiO	[22]	$O(\kappa^5\epsilon^{-2})$	$O(\kappa^9\epsilon^{-2})$	$O(\kappa^5\epsilon^{-2})$	$\tilde{O}(\kappa^6\epsilon^{-2})$	
SBiO-BreD	Ours	$O(\kappa^5\epsilon^{-2})$	$O(\kappa^5\epsilon^{-2})$	$O(\kappa^5\epsilon^{-2})$	$\tilde{O}(\kappa^6\epsilon^{-2})$	√
ASBiO-BreD	Ours	$O(\kappa^5\epsilon^{-1.5})$	$O(\kappa^5\epsilon^{-1.5})$	$O(\kappa^5\epsilon^{-1.5})$	$\tilde{O}(\kappa^6\epsilon^{-1.5})$	√

$$\min_{x\in\mathcal{X}\subseteq\mathbb{R}^{d_1}} \mathbb{E}_{\xi\sim\mathcal{D}}\big[f(x,y^*(x);\xi)\big] + h(x),$$

$$\text{s.t. } y^*(x) \in \arg\min_{y\in\mathbb{R}^{d_2}} \mathbb{E}_{\zeta\sim\mathcal{D}'}\big[g(x,y;\zeta)\big],$$

Roadmap

- Background
- Enhanced Bilevel Optimization via Bregman Distance
- Convergence Properties
- Experimental Results
- Conclusions

Experimental Results

(1) Data Hyper-cleaning

$$\min_{\lambda} \; l_{val}(\lambda, w^*(\lambda)) := \frac{1}{|D_{\mathcal{V}}|} \sum_{(x_i, y_i) \in D_{\mathcal{V}}} l(x_i^T w^*(\lambda), y_i)$$

$$\text{s.t.} \;\; w^*(\lambda) = \arg\min_{w} \; l_{tr}(\lambda, w) := \frac{1}{|D_{\mathcal{T}}|} \sum_{(x_i, y_i) \in D_{\mathcal{T}}} \sigma(\lambda_i) l(x_i^T w, y_i) + C\|w\|^2,$$

Experimental Results

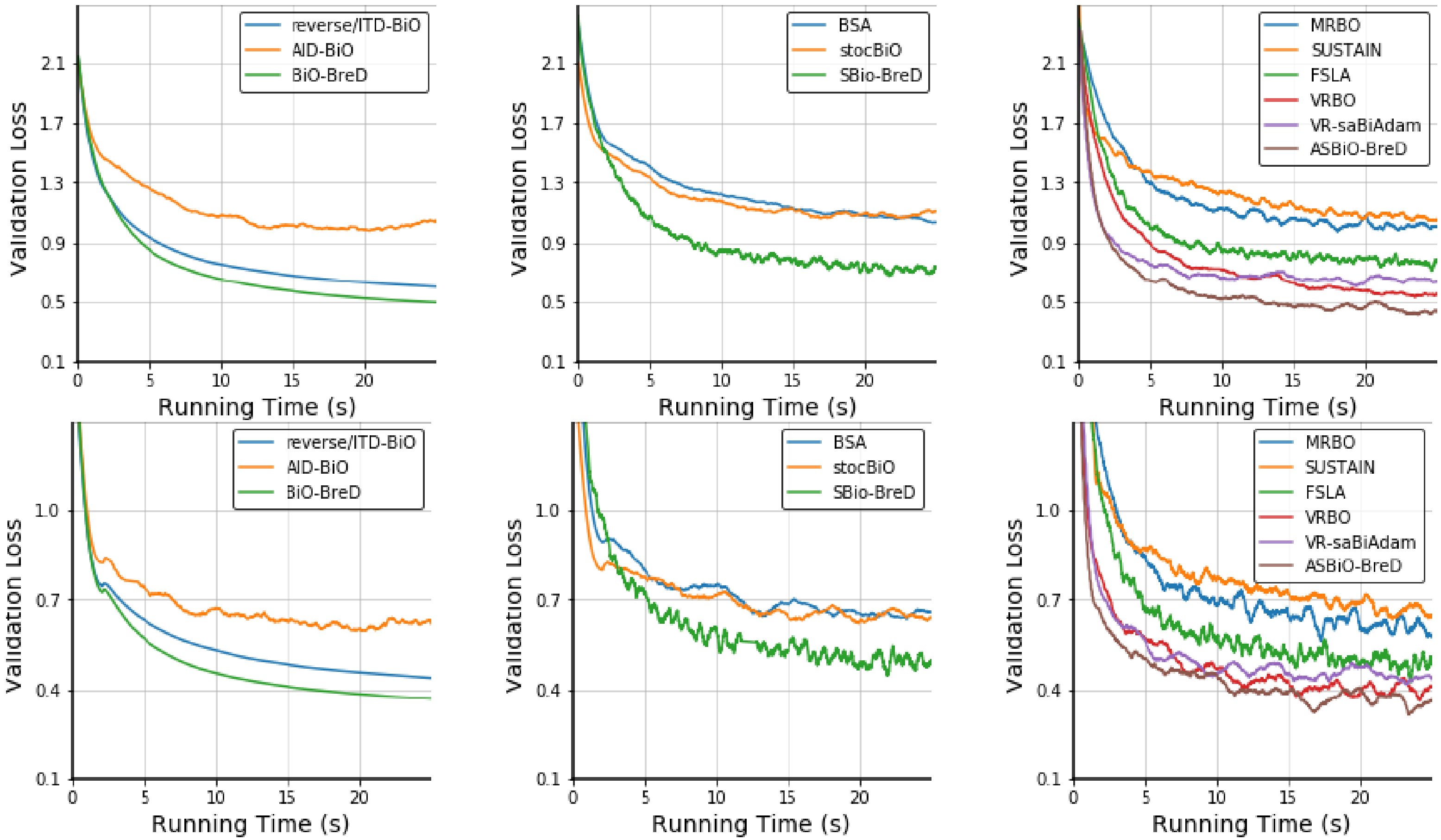

Figure 1: Validation Loss vs. Running Time for different methods. We compare our BiO-BreD with deterministic baselines (the first column), SBiO-BreD with stochastic baselines (the second column); ASBiO-BreD with momentum-based or SPIDER/SARAH based baselines (the last column). We test two values of ϱ: large noise setting $\varrho = 0.8$ (top row) and small noise setting $\varrho = 0.4$ (bottom row).

Experimental Results

(2) Hyper-representation Learning

$$\min_{\lambda}\ l_{val}(\lambda, w^*(\lambda)) := \mathbb{E}_{\xi}\Big[\frac{1}{|D_{\mathcal{V},\xi}|}\sum_{(x_i,y_i)\in D_{\mathcal{V},\xi}} l\Big(w_{\xi}^*(\lambda)^T\phi(x_i;\lambda), y_i\Big);\xi\Big] + \alpha\|\lambda\|_1$$

$$\text{s.t. } w_{\xi}^*(\lambda) = \arg\min_{w}\ l_{tr}(\lambda, w;\xi) := \frac{1}{|D_{\mathcal{T},\xi}|}\sum_{(x_i,y_i)\in D_{\mathcal{T},\xi}} l\Big(w^T\phi(x_i;\lambda), y_i\Big) + C\|w\|^2,$$

Experimental Results

Table 3: Validation accuracy *vs.* Running Time (5-way-1-shot) for different methods (with L_1 regularization)

Time	AID_BiO	ITD_BiO	MRBO	FSLA	VRBO	VR-saBiAdam	ASBiO-BreD
20s	0.6509	0.6411	0.6103	0.6539	0.5951	0.6812	0.6653
40s	0.7365	0.7210	0.6971	0.7399	0.6805	0.7141	0.7403
60s	0.7762	0.7721	0.7519	0.7661	0.7429	0.7523	**0.7830**

Table 4: Validation accuracy *vs.* Running Time (5-way-5-shot) for different methods (with L_1 regularization)

Time	AID_BiO	ITD_BiO	MRBO	FSLA	VRBO	VR-saBiAdam	ASBiO-BreD
20s	0.8316	0.8131	0.8174	0.7993	0.7730	0.7753	0.8529
40s	0.8779	0.8621	0.8634	0.8485	0.8305	0.8188	0.8967
60s	0.9032	0.8968	0.8819	0.8824	0.8745	0.8640	**0.9313**

Roadmap

- Background
- Enhanced Bilevel Optimization via Bregman Distance
- Convergence Properties
- Experimental Results
- Conclusions

Conclusions

- 1) We proposed a class of enhanced bilevel optimization methods based on Bregman distance.

- 2) We provided a comprehensive convergence analysis framework for our methods, and proved that our methods achieve a lower computational complexity than the best known results.

Thanks!

Q&A