Value Function based Difference-of-Convex Algorithm for Bilevel Hyperparameter Selection Problems

Lucy Gao, Jane J. Ye, Haian Yin, Shangzhi Zeng, Jin Zhang

ICML | 2022
Thirty-ninth International Conference on Machine Learning

Hyperparameter Bilevel Program(BLP)

- We consider the following BLP framework:

$$\begin{aligned} \min_{x \in \mathbb{R}^n, \lambda \in \mathbb{R}^J_+} \quad & L(x) \\ \text{s.t.} \quad & x \in \operatorname*{argmin}_{x' \in \mathbb{R}^n} \left\{ l(x') + \sum_{i=1}^{J} \lambda_i P_i(x') \right\} \end{aligned}$$

- λ is a vector of hyperparameters
- $L : \mathbb{R}^n \to \mathbb{R}$ is the convex function for the validation error
- $l : \mathbb{R}^n \to \mathbb{R}$ is the convex function for the training error
- $P_i : \mathbb{R}^n \to \mathbb{R}_+, \ i = 1, \ldots, J$ are convex regularizers

Examples of hyperparameter BLPs

Machine learning algorithm	x	λ	$L(x)/l(x)$	$\sum_{i=1}^{J} \lambda_i P_i(x)$
elastic net	$\boldsymbol{\beta}$	$\lambda_1,\ \lambda_2$	$\frac{1}{2}\sum_{i\in I_{\text{val}}/i\in I_{\text{tr}}} \lvert b_i - \boldsymbol{\beta}^\top \mathbf{a}_i \rvert^2$	$\lambda_1 \lVert\boldsymbol{\beta}\rVert_1 + \frac{\lambda_2}{2}\lVert\boldsymbol{\beta}\rVert_2^2$
sparse group lasso	$\boldsymbol{\beta}$	$\lambda \in \mathbb{R}_+^{M+1}$	$\frac{1}{2}\sum_{i\in I_{\text{val}}/i\in I_{\text{tr}}} \lvert b_i - \boldsymbol{\beta}^\top \mathbf{a}_i \rvert^2$	$\sum_{m=1}^{M} \lambda_m \lVert\boldsymbol{\beta}^{(m)}\rVert_2 + \lambda_{M+1}\lVert\boldsymbol{\beta}\rVert_1$
low-rank matrix completion	$\boldsymbol{\theta},\ \boldsymbol{\beta},\ \Gamma$	$\lambda \in \mathbb{R}_+^{2G+1}$	$\sum_{(i,j)\in\Omega_{\text{val}}/(i,j)\in\Omega_{\text{tr}}} \frac{1}{2}\lvert M_{ij} - \mathbf{x}_i\boldsymbol{\theta} - \mathbf{z}_j\boldsymbol{\beta} - \Gamma_{ij}\rvert^2$	$\lambda_0\lVert\Gamma\rVert_* + \sum_{g=1}^{G}\lambda_g\lVert\boldsymbol{\theta}^{(g)}\rVert_2 + \sum_{g=1}^{G}\lambda_{g+G}\lVert\boldsymbol{\beta}^{(g)}\rVert_2$
support vector machine	$\mathbf{w},\ c$	$\lambda,\ \bar{\mathbf{w}}$	$\sum_{j\in I_{\text{val}}/j\in I_{\text{tr}}} \max(1 - b_j(\mathbf{w}^\top\mathbf{a}_j - c), 0)$	$\frac{\lambda}{2}\lVert\mathbf{w}\rVert^2$

Hyperparameter Decoupling

- The Lower Level (LL) problem

$$\min_{x'} \; l(x') + \sum_{i=1}^{J} \lambda_i P_i(x').$$

- The hyperparameter variables λ can be decoupled from the regularization term by introducing a new variable r

$$\min_{x'} \; l(x') \quad \text{s.t. } P_i(x') \leq r_i, \; i = 1, \ldots, J.$$

- This suggests working with the following BLP:

$$\begin{aligned} \min_{x, r \in \mathbb{R}^J_+} \quad & L(x) \\ \text{s.t.} \quad & x \in \arg\min_{x'} \{ l(x') \text{ s.t. } P_i(x') \leq r_i, i = 1, \ldots, J \}. \end{aligned}$$

Single-level DC Reformulation

- The value function of the LL problem

$$v(r) := \min \{l(x) \text{ s.t. } P_i(x) \le r_i, \ i = 1, \dots, J\}.$$

- Thanks to full convexity, $v(r)$ is convex.
- Using the value function, we can reformulate BLP as the following Difference-of-Convex(DC) program:

$$\begin{aligned} \min_{x, r \in \mathbb{R}^J_+} \quad & L(x) \\ \text{s.t.} \quad & l(x) - v(r) \le 0, P_i(x) \le r_i, \ i = 1, \dots, J. \end{aligned}$$

VF-iDCA

- Given a current iterate (x^k, r^k) for each k, solving the LL problem parameterized by r^k

$$\tilde{x}^k \in \arg\min_x \quad l(x) \text{ s.t. } P_i(x) \leq r_i^k, \;\; i = 1, \ldots, J,$$

- Find a corresponding Karush-Kuhn-Tucker (KKT) multiplier γ^k.
- Construct a linearization of $v(r)$ at r^k,

$$V_k(x, r) := l(x) - l(\tilde{x}^k) + \langle \gamma^k, r - r^k \rangle.$$

- Update $z^{k+1} := (x^{k+1}, r^{k+1})$ by solving the strongly convex subproblem

$$\min_{x, r \in \mathbb{R}_+^J} \quad \phi_k(x, r) := L(x) + \frac{\rho}{2}\|z - z^k\|^2 + \alpha_k \max_{i=1,\ldots,J} \{0, V_k(x, r), P_i(x) - r_i\},$$

where $\rho > 0$, and α_k represents the adaptive penalty parameter, $z := (x, r)$, $z^k := (x^k, r^k)$

Theoretical Investigations

Theorem . *Assume that* $L(x)$, $l(x)$ *and* $P(x)$ *are semi-algebraic functions. Suppose that* $\{z^k := (x^k, r^k)\}$ *and* $\{\alpha_k\}$ *generated by VF-iDCA are bounded,* $L(x)$ *is bounded below and there exists* $\delta > 0$ *such that* $r_i^k \geq \delta$ *for all* k *and* $i = 1, \ldots, J$. *Then* $\{z^k\}$ *converges to a KKT point of DC program.*

Numerical Experiments on Synthetic Data

Table 1. Elastic net problems on synthetic data.

Settings	Method	Time	Val. Err.	Test Err.
$\lvert I_{tr}\rvert = 100$, $\lvert I_{val}\rvert = 20$, $\lvert I_{test}\rvert = 250$, $p = 250$	Grid	3.10 ± 0.44	6.16 ± 2.35	6.68 ± 1.16
	Random	3.55 ± 0.58	5.98 ± 2.24	6.67 ± 1.15
	TPE	5.41 ± 0.75	6.05 ± 2.30	6.77 ± 1.04
	IGJO	2.04 ± 1.46	4.43 ± 1.77	5.13 ± 1.37
	IFDM	1.33 ± 0.55	4.41 ± 0.96	4.77 ± 1.46
	VF-iDCA	**0.91 ± 0.19**	1.95 ± 0.81	**3.99 ± 0.69**
$\lvert I_{tr}\rvert = 100$, $\lvert I_{val}\rvert = 100$, $\lvert I_{test}\rvert = 250$, $p = 250$	Grid	3.17 ± 0.43	6.51 ± 1.53	6.82 ± 1.10
	Random	5.29 ± 0.60	6.44 ± 1.53	6.77 ± 1.14
	TPE	5.40 ± 0.84	6.44 ± 1.53	6.76 ± 1.06
	IGJO	2.42 ± 1.30	4.71 ± 1.32	4.88 ± 1.30
	IFDM	1.30 ± 0.41	4.78 ± 1.12	4.61 ± 1.12
	VF-iDCA	**1.37 ± 0.29**	3.04 ± 0.74	**3.58 ± 0.60**
$\lvert I_{tr}\rvert = 100$, $\lvert I_{val}\rvert = 100$, $\lvert I_{test}\rvert = 100$, $p = 2500$	Grid	19.05 ± 1.63	7.95 ± 1.10	8.54 ± 0.81
	Random	35.42 ± 3.55	7.90 ± 1.09	8.52 ± 0.79
	TPE	32.17 ± 7.40	7.89 ± 1.11	8.60 ± 0.87
	IGJO	16.12 ± 40.95	7.99 ± 1.18	8.41 ± 0.86
	IFDM	**4.38 ± 2.53**	7.97 ± 0.83	8.53 ± 1.53
	VF-iDCA	19.97 ± 5.17	1.61 ± 1.85	**5.10 ± 1.07**

Table 2. Sparse group lasso problems on synthetic data.

Settings	Method	#λ	Time	Val. Err.	Test Err.
$p = 600$, $M = 30$	Grid	2	30.38 ± 1.82	42.45 ± 7.67	44.56 ± 7.33
	Random	31	28.54 ± 1.51	39.27 ± 7.32	43.00 ± 8.83
	TPE	31	47.07 ± 4.01	35.69 ± 5.92	40.59 ± 6.67
	IGJO	31	69.62 ± 47.76	30.16 ± 7.41	39.28 ± 6.56
	VF-iDCA	31	**8.13 ± 1.20**	0.01 ± 0.00	**38.50 ± 6.00**
$p = 600$, $M = 300$	Grid	2	20.84 ± 1.04	41.88 ± 7.64	44.90 ± 7.02
	Random	301	18.94 ± 1.09	43.92 ± 8.77	47.90 ± 8.55
	TPE	301	76.82 ± 2.55	39.22 ± 6.26	42.93 ± 8.00
	IGJO	301	160.85 ± 71.50	20.37 ± 4.46	38.52 ± 6.78
	VF-iDCA	301	56.73 ± 92.48	19.61 ± 8.33	**33.55 ± 4.71**
$p = 1200$, $M = 300$	Grid	2	87.20 ± 5.85	49.56 ± 10.76	51.85 ± 12.90
	Random	301	73.75 ± 4.28	53.65 ± 12.03	55.84 ± 14.25
	TPE	301	117.07 ± 5.66	45.94 ± 9.30	51.67 ± 12.29
	IGJO	301	98.35 ± 47.47	20.70 ± 4.70	38.90 ± 7.20
	VF-iDCA	301	**23.41 ± 1.31**	17.90 ± 3.47	**36.90 ± 7.48**

Table 3. Low-rank matrix completion problems on synthetic data.

Method	#λ	Time	Val. Err.	Test Err.
Grid	2	20.67 ± 0.90	0.71 ± 0.21	0.76 ± 0.20
Random	25	32.49 ± 1.84	0.73 ± 0.21	0.80 ± 0.20
TPE	25	35.05 ± 9.37	0.68 ± 0.20	0.76 ± 0.18
IGJO	25	1268.65 ± 365.99	0.68 ± 0.21	0.72 ± 0.18
VF-iDCA	25	51.55 ± 10.43	0.06 ± 0.07	**0.70 ± 0.16**

Competitors:

- **Implicit Differentiation**: **IGJO** (Feng & Simon, 2018) and **IFDM** (Bertrand et al., 2020).
- **Grid Search**
- **Random Search**
- **TPE**: Tree-structured Parzen Estimator approach (Bergstra et al., 2013)

Application to real data

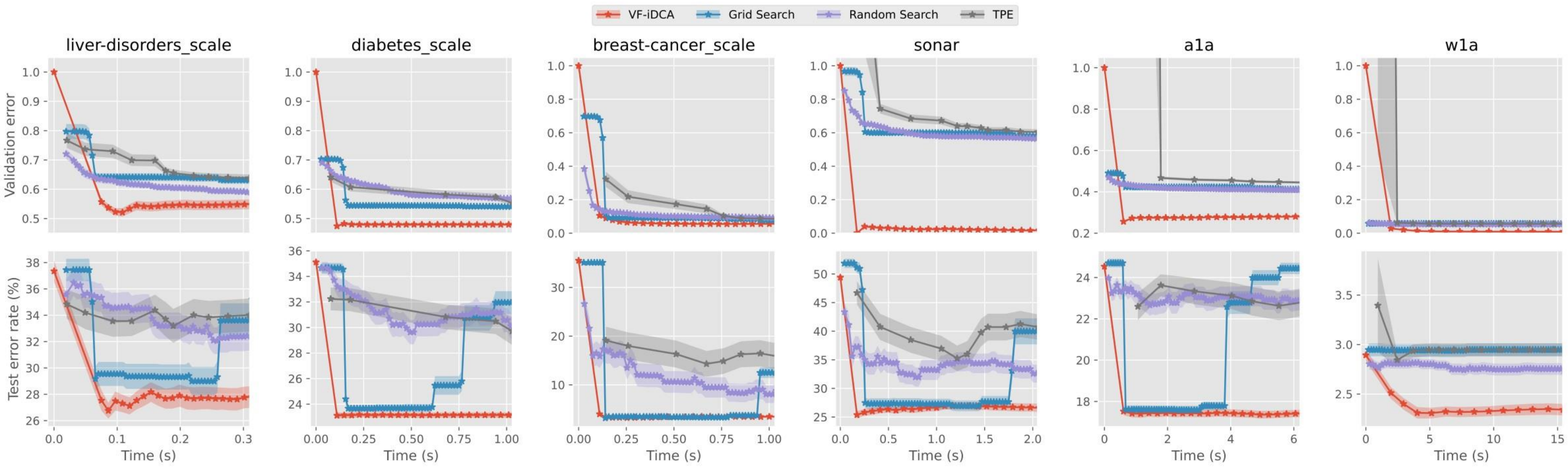

Figure 1. Comparison of the algorithms on SVM problem (validation error and test error versus time) for 6 datasets: liver-disorders_scale, diabetes_scale, breast-cancer_scale, sonar, a1a, w1a

Competitors:

- **Grid Search**
- **Random Search**
- **TPE**: Tree-structured Parzen Estimator approach (Bergstra et al., 2013)

Thanks for your attention

Code is available at

https://github.com/SUSTech-Optimization/VF-iDCA