Virtual Conference, Feb 2021

Enhancing Parameter-Free Frank Wolfe with an Extra Subproblem

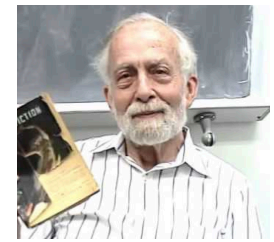
Bingcong Li,[*] Lingda Wang,[#] Georgios B. Giannakis,[*] and Zhizhen Zhao[#]

[*]University of Minnesota, Twin Cities
[#]University of Illinois at Urbana-Champaign

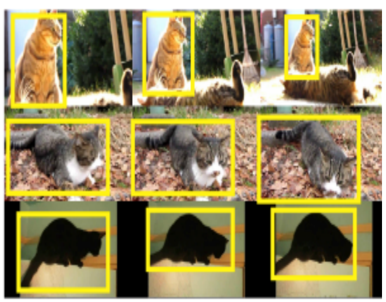
Acknowledgement: NSF 1711471, 1901134, and Alfred P. Sloan Foundation

Context and motivation

- **F**rank-**W**olfe (conditional gradient) method
  - Invented by to M. Frank and P. Wolfe in 1956
  - Constrained **convex** optimization (this talk)
  - Low iteration complexity, sparse-promoting



- Applications



video colocation
[Joulin et al '14]

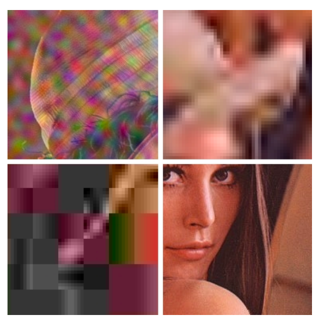
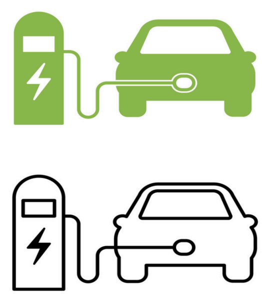
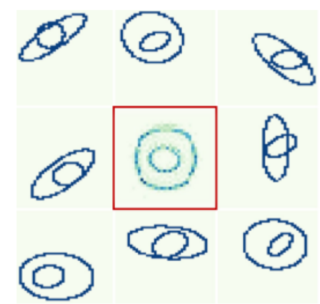


image reconstruction
[Harchaoui et al '15]



EV charging
[Zhang & GG '18]

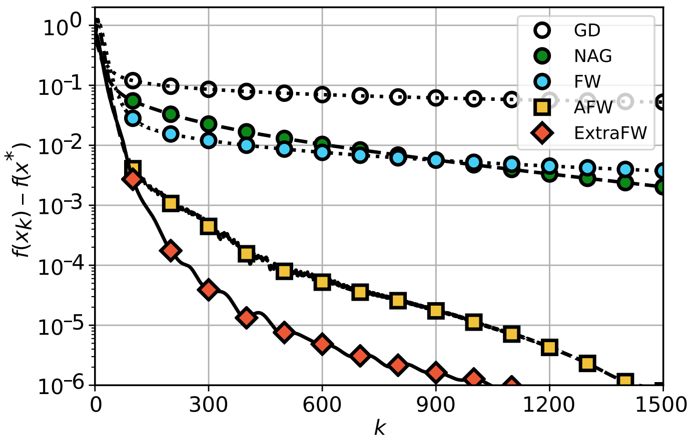


optimal transport
[Luise et al '19]

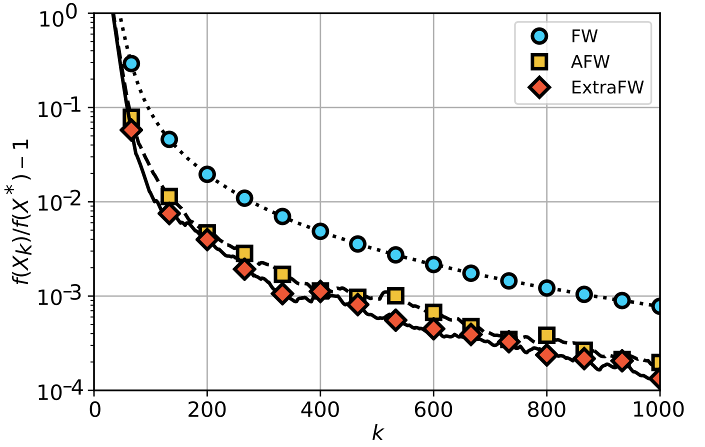
M. Frank, P. Wolfe,. "An algorithm for quadratic programming". *Naval research logistics quarterly*. 1956 Mar;3(1-2):95-110.

Contributions in a nutshell

- Faster FW **without** problem-dependent parameters
  - Impossible in general: FW is lower-bound-matching
  - ExtraFW converges faster on certain constraints
  - with simple step size $\mathcal{O}\left(\frac{1}{k}\right)$
- Promising numerical performance



binary classification



matrix completion

Roadmap

- **Preliminaries**
- Algorithm design and analysis
- Numerical experiments

Problem statement

- Objective and constraint

$$\min_{\mathbf{x}\in\mathcal{X}} f(\mathbf{x})$$

Assumption 1. (Lipschitz Continuous Grad.) $\|\nabla f(\mathbf{x}) - \nabla f(\mathbf{y})\|_* \leq L\|\mathbf{x}-\mathbf{y}\|$

Assumption 2. (Convex Objective Function.) $f(\mathbf{y}) - f(\mathbf{x}) \geq \langle \nabla f(\mathbf{x}), \mathbf{y}-\mathbf{x}\rangle$

Assumption 3. (Convex and Compact Constraint.) $\|\mathbf{x}-\mathbf{y}\| \leq D, \ \forall \mathbf{x},\mathbf{y}\in\mathcal{X}$

- **Goal:** solve this problem with neither projection nor L
  - FW variants eliminate projection
  - L estimate is usually too pessimistic
  - L related step sizes do not perform that well empirically

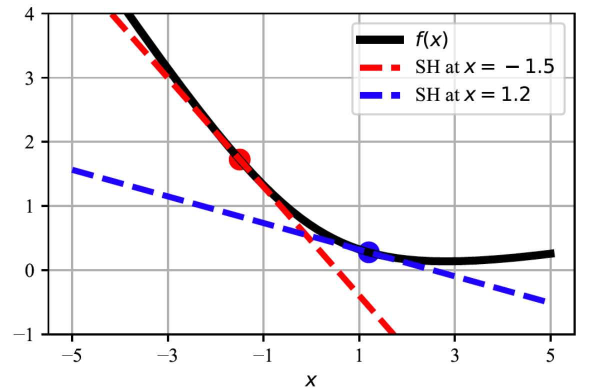
FW recap

- FW's geometry and convergence

From *k* = *0,* iteratively update via

$$\mathbf{v}_{k+1} = \arg\min_{\mathbf{x}\in\mathcal{X}} \langle \nabla f(\mathbf{x}_k), \mathbf{x} \rangle$$

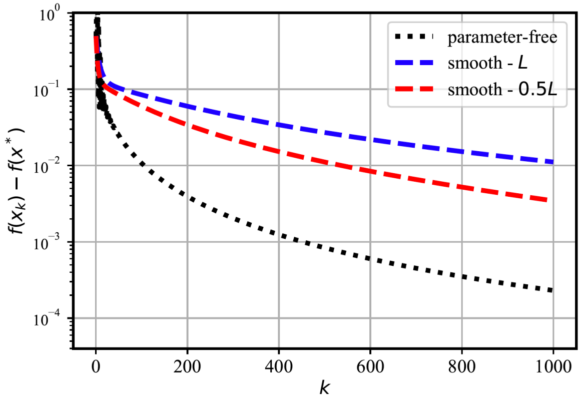
$$\mathbf{x}_{k+1} = (1-\delta_k)\mathbf{x}_k + \delta_k \mathbf{v}_{k+1}$$



- Geometry: $\mathbf{v}_{k+1} = \arg\min_{\mathbf{x}\in\mathcal{X}} f(\mathbf{x}_k) + \langle \nabla f(\mathbf{x}_k), \mathbf{x} - \mathbf{x}_k \rangle$
- Convergence: $f(\mathbf{x}_k) - f(\mathbf{x}^*) = \mathcal{O}(\frac{LD^2}{k})$
  - ✓ Parameter-free step size $\delta_k = \frac{2}{k+2}$
  - ✓ Smooth step size $\delta_k = \min\left\{\frac{\langle \nabla f(\mathbf{x}_k), \mathbf{x}_k - \mathbf{v}_{k+1}\rangle}{L\|\mathbf{v}_{k+1} - \mathbf{x}_k\|^2}, 1\right\}$
  - ✓ Line search (function evaluation needed)

Faster FW (variants)

- Smooth step sizes / line search
  - **FW** [Levitin & Polyak 1966]: active and strongly convex $\mathcal{X}$
  - **FW** [Garber & Hazan '15]: strongly convex f , strongly convex $\mathcal{X}$
  - **Away-steps** [L.-Julien & Jaggi '15]: strongly convex f, polytope $\mathcal{X}$



- Parameter-free step sizes
  - Challenges: $f(\mathbf{x}_{k+1}) \leq f(\mathbf{x}_k)$ is not guaranteed using this step size
  - **FW** [Bach '20]: twice differentiable f, polytope $\mathcal{X}$
  - **AFW** [Li et al '20]: replacing NAG subproblem with a FW subproblem

- Other faster FW variants
  - **CGS** [Lan & Zhou, '16]: replacing NAG subproblem with CGS
  - Relies on both L and D

Roadmap

- ❑ Preliminaries
- ❑ **Algorithm design and analysis**
- ❑ Numerical experiments

ExtraFW: update via prediction - correction

From $k = 0$, iteratively update via

$$\mathbf{y}_k = (1-\delta_k)\mathbf{x}_k + \delta_k \mathbf{v}_k$$

$$\hat{\mathbf{g}}_{k+1} = (1-\delta_k)\mathbf{g}_k + \delta_k \nabla f(\mathbf{y}_k)$$

$$\hat{\mathbf{v}}_{k+1} = \arg\min_{\mathbf{v}\in\mathcal{X}} \langle \hat{\mathbf{g}}_{k+1}, \mathbf{v}\rangle$$

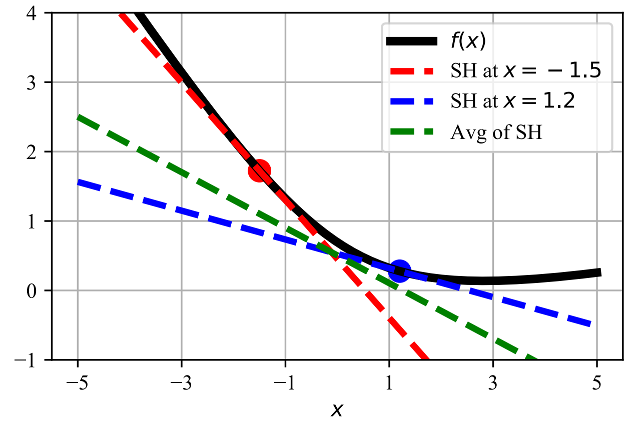
prediction

$$\mathbf{x}_{k+1} = (1-\delta_k)\mathbf{x}_k + \delta_k \hat{\mathbf{v}}_{k+1}$$

$$\mathbf{g}_{k+1} = (1-\delta_k)\mathbf{g}_k + \delta_k \nabla f(\mathbf{x}_{k+1})$$

$$\mathbf{v}_{k+1} = \arg\min_{\mathbf{v}\in\mathcal{X}} \langle \mathbf{g}_{k+1}, \mathbf{v}\rangle$$

correction



- Lower bound prediction $\hat{\mathbf{g}}_{k+1} = \sum_{\tau=0}^{k-1} w_k^\tau \nabla f(\mathbf{x}_{\tau+1}) + \delta_k \nabla f(\mathbf{y}_k)$

$$\hat{\mathbf{v}}_{k+1} = \text{argmin}_{\mathbf{x}\in\mathcal{X}} \sum_{\tau=0}^{k-1} w_k^\tau \Big[f(\mathbf{x}_{\tau+1}) + \langle \nabla f(\mathbf{x}_{\tau+1}), \mathbf{x} - \mathbf{x}_{\tau+1}\rangle \Big] + \boxed{\delta_k \Big[f(\mathbf{y}_k) + \Big\langle \nabla f(\mathbf{y}_k), \mathbf{x} - \mathbf{y}_k \Big\rangle \Big]}$$

- Lower bound correction $\mathbf{g}_{k+1} = \sum_{\tau=0}^{k-1} w_k^\tau \nabla f(\mathbf{x}_{\tau+1}) + \delta_k \nabla f(\mathbf{x}_{k+1})$

$$\mathbf{v}_{k+1} = \text{argmin}_{\mathbf{x}\in\mathcal{X}} \sum_{\tau=0}^{k-1} w_k^\tau \Big[f(\mathbf{x}_{\tau+1}) + \langle \nabla f(\mathbf{x}_{\tau+1}), \mathbf{x} - \mathbf{x}_{\tau+1}\rangle \Big] + \boxed{\delta_k \Big[f(\mathbf{x}_{k+1}) + \Big\langle \nabla f(\mathbf{x}_{k+1}), \mathbf{x} - \mathbf{x}_{k+1} \Big\rangle \Big]}$$

Convergence of ExtraFW

- For general problems with

Theorem: Let $\mathbf{g}_0 = \mathbf{0}$ and $\delta_k = \frac{2}{k+3}$, then ExtraFW guarantees that

$$f(\mathbf{x}_k) - f(\mathbf{x}^*) = \mathcal{O}\Big(\frac{LD^2}{k}\Big)$$

- What prevents a faster rate?

  Difficulties to bound $\|\mathbf{v}_k - \hat{\mathbf{v}}_k\|^2$ due to non-uniqueness of $\mathbf{v}_k$

- Faster rates on active norm ball constraints

Assumption 4. The constraint is active

- Common in machine learning problems

$$\min_{\mathbf{x}} f(\mathbf{x}) \;\; \text{s.t.} \;\; g(\mathbf{x}) \leq R \quad \Leftrightarrow \quad \min_{\mathbf{x}} f(\mathbf{x}) + \gamma g(\mathbf{x})$$

- Closed-form solution of $\mathbf{v}_k$ makes small $\|\mathbf{v}_k - \hat{\mathbf{v}}_k\|^2$ possible

Acceleration of ExtraFW

constraint	**ExtraFW**	**AFW**
$\|\mathbf{x}\|_2 \leq R$	$\mathcal{O}\Big(\min\Big\{\frac{LD^2}{k}, \frac{LD^2T}{k^2}\Big\}\Big)$	$\mathcal{O}\Big(\min\Big\{\frac{LD^2}{k}, \frac{LD^2T\ln k}{k^2}\Big\}\Big)$
$\|\mathbf{x}\|_1 \leq R$	$\mathcal{O}\Big(\min\Big\{\frac{LD^2}{k}, \frac{LD^2T}{k^2}\Big\}\Big)$	$\mathcal{O}\Big(\min\Big\{\frac{LD^2}{k}, \frac{LD^2T}{k^2}\Big\}\Big)$
$\|\mathbf{x}\|_{\mathrm{n-sp}} \leq R$	$\mathcal{O}\Big(\min\Big\{\frac{LD^2}{k}, \frac{LD^2T}{k^2}\Big\}\Big)$	$\mathcal{O}\Big(\min\Big\{\frac{LD^2}{k}, \frac{LD^2T\ln k}{k^2}\Big\}\Big)$

- Local acceleration: after T iterations, the bound is improved over FW

❑ Remarks

- Implementation is the same regardless of acceleration
- Merits of PC update: improved k dependence over AFW
- Not too many algorithms achieve (local) acceleration without relying on L
- Extendable to Frobenius and the nuclear norm ball constraints

B. Li, M. Coutino, G. B. Giannakis, and G. Leus. “How Does Momentum Help Frank Wolfe?” 2020, *arXiv:2006.11116*.

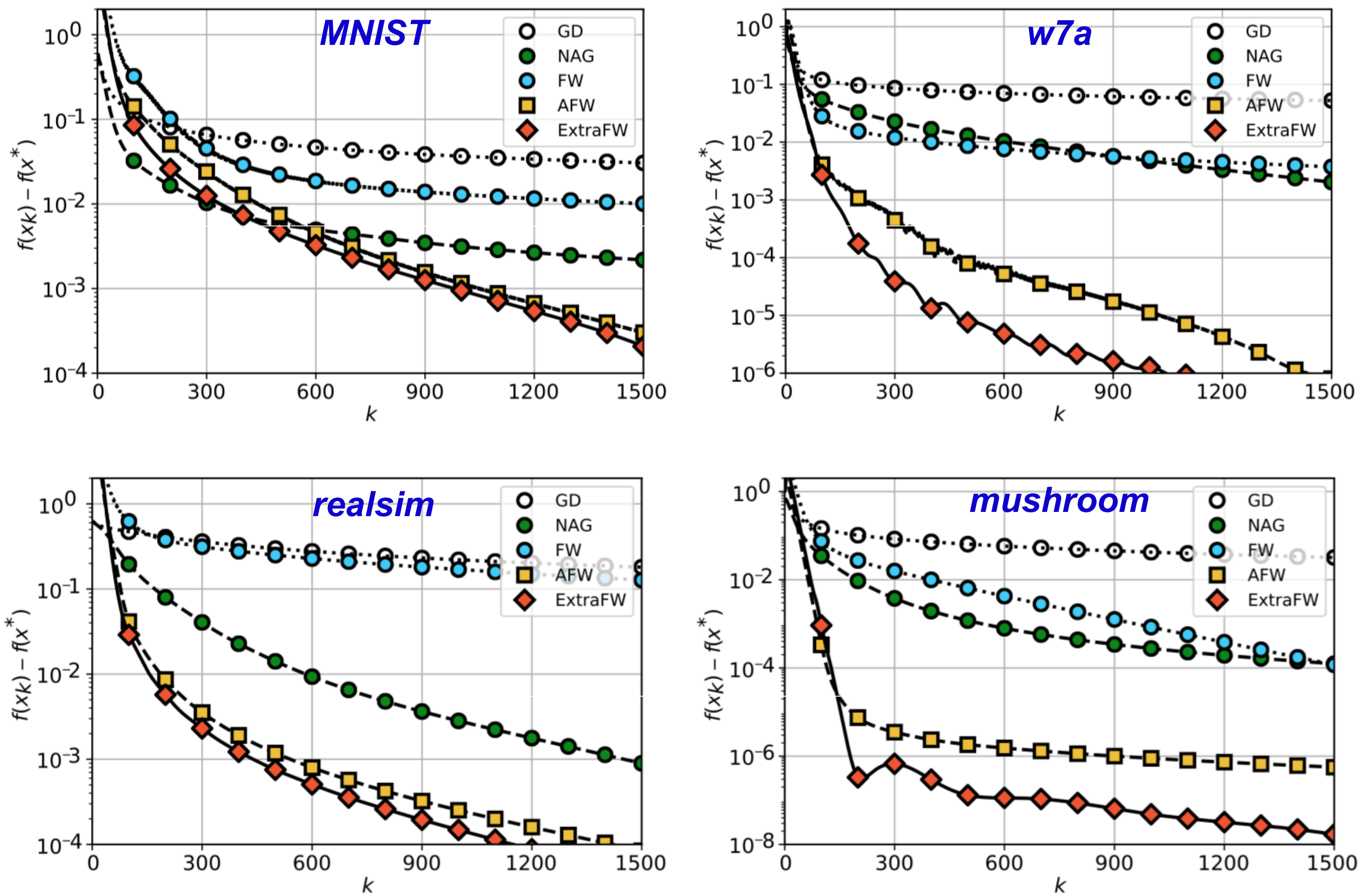
Roadmap

- ❑ Preliminaries

- ❑ Algorithm design and analysis
- ❑ **Numerical experiments**

Binary classification

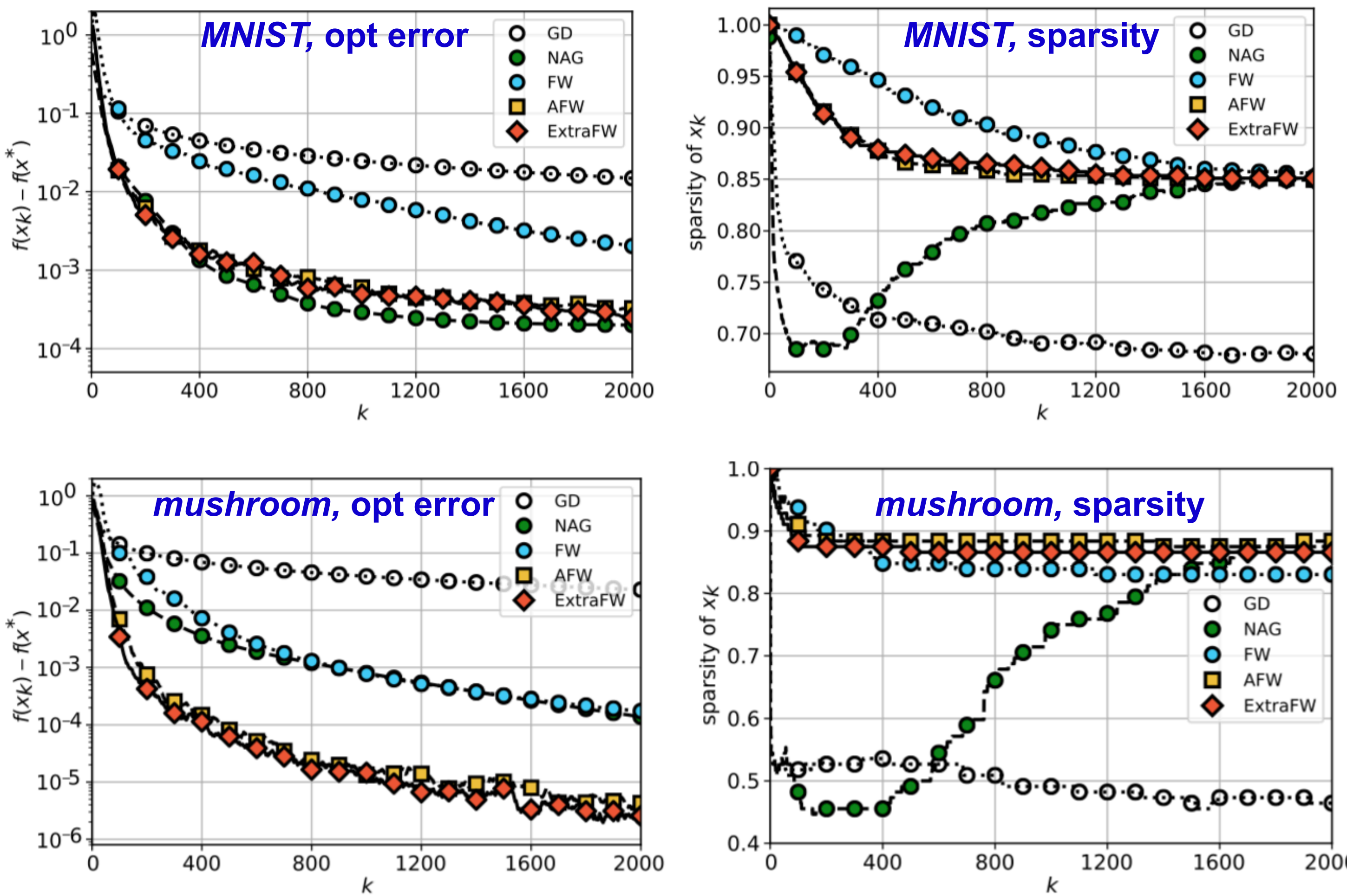
$$f(\mathbf{x}) = \frac{1}{N}\sum_{i=1}^{N} \ln\left(1+\exp(-b_i\langle \mathbf{a}_i, \mathbf{x}\rangle)\right) \qquad \mathcal{X} = \{\mathbf{x} | \|\mathbf{x}\|_2 \le R\}$$



Datasets from LIBSVM https://www.csie.ntu.edu.tw/~cjlin/libsvmtools/datasets/binary.html
and MNIST http://yann.lecun.com/exdb/mnist/ are adopted.

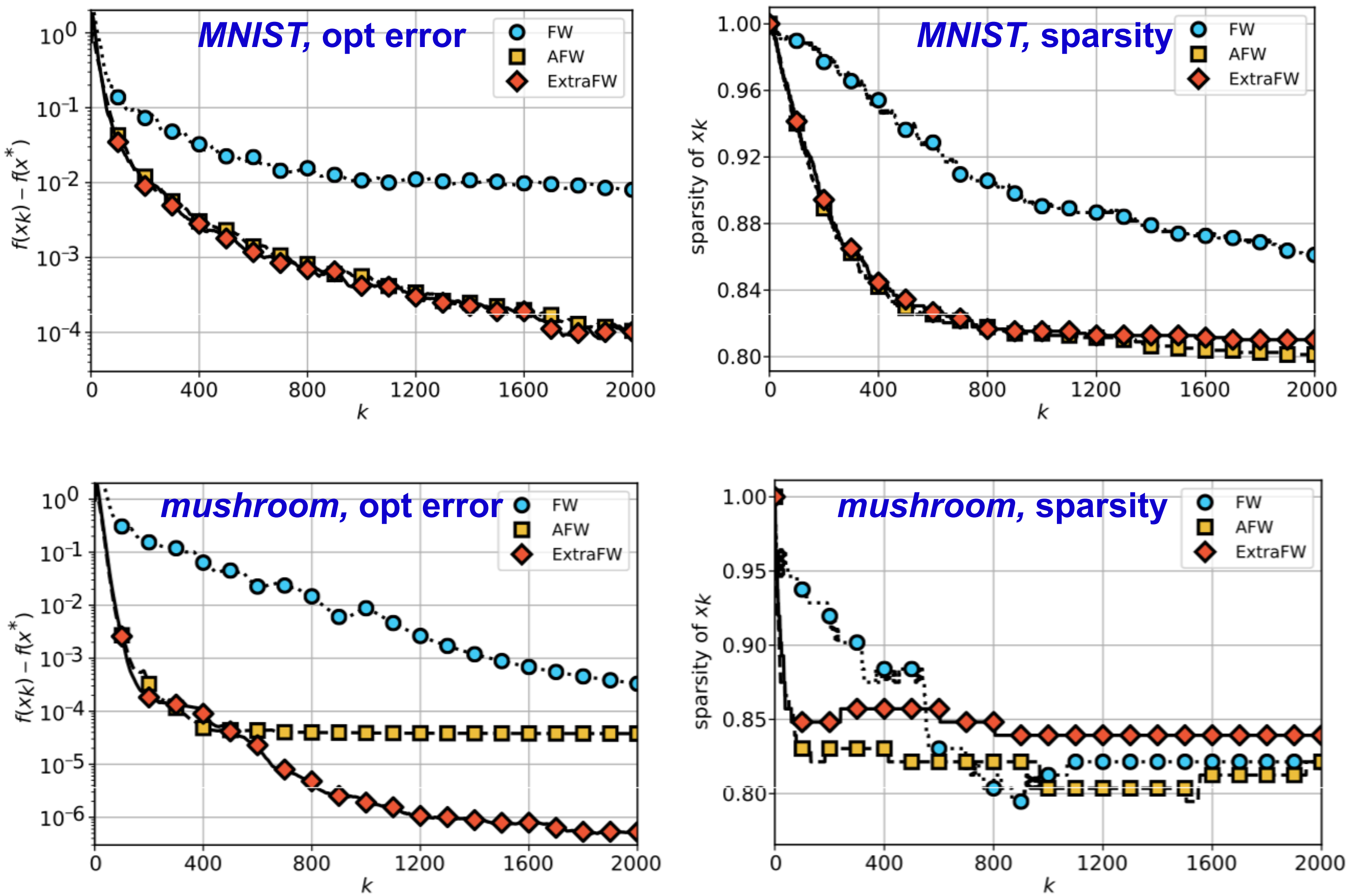
Binary classification

$$f(\mathbf{x}) = \frac{1}{N}\sum_{i=1}^{N} \ln\left(1+\exp(-b_i\langle \mathbf{a}_i, \mathbf{x}\rangle)\right) \qquad \mathcal{X} = \{\mathbf{x} \,|\, \|\mathbf{x}\|_1 \le R\}$$



Binary classification

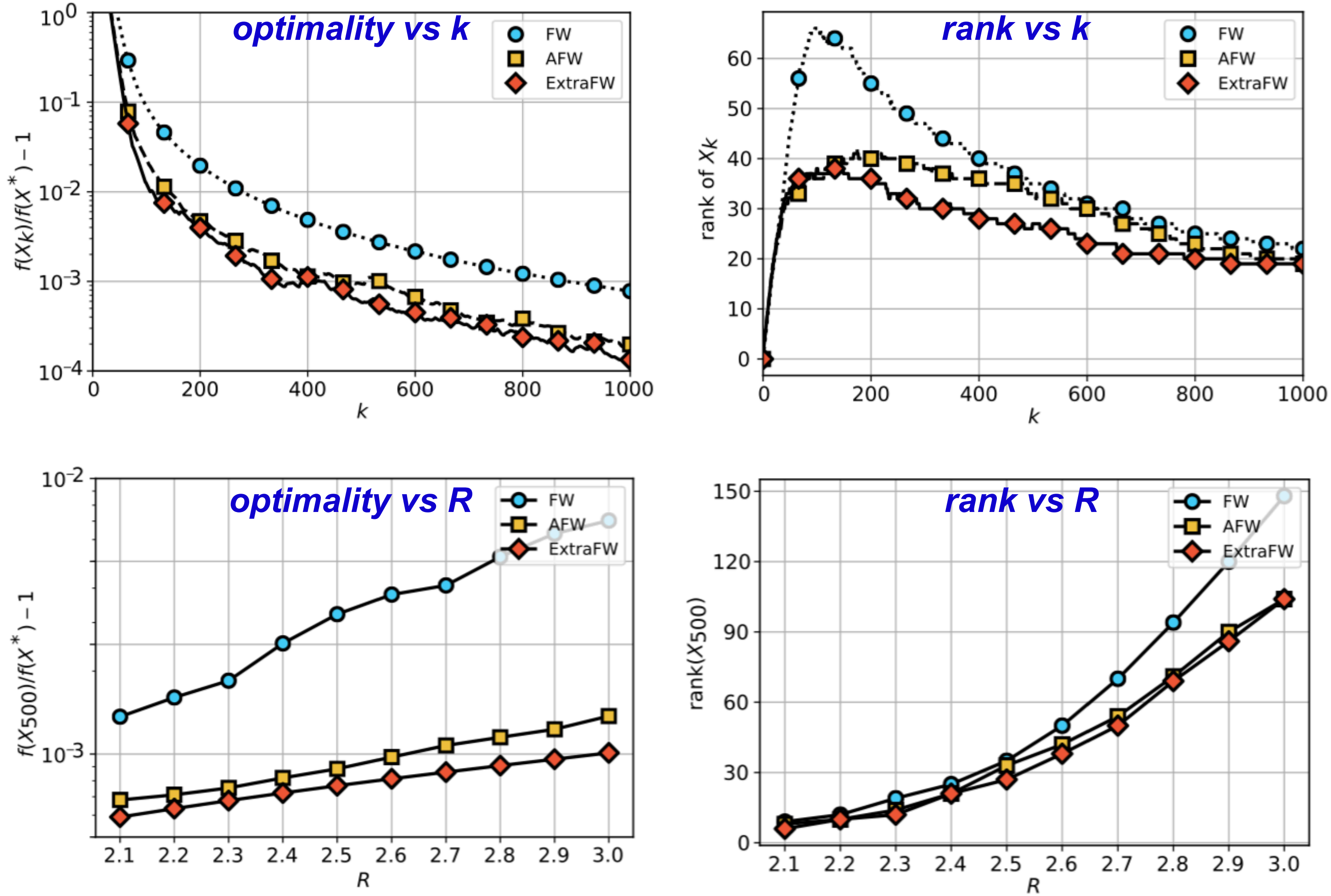
$$f(\mathbf{x}) = \frac{1}{N}\sum_{i=1}^{N} \ln\left(1 + \exp(-b_i\langle \mathbf{a}_i, \mathbf{x}\rangle)\right) \qquad \mathcal{X} = \mathrm{conv}\{\mathbf{x} \,|\, \|\mathbf{x}\|_0 \leq n, \|\mathbf{x}\|_2 \leq R\}$$



Argyriou, A., Foygel, R. and Srebro, N., 2012. Sparse Prediction with the k-Support Norm.
In *Advances in Neural Information Processing Systems* (pp. 1457-1465).

Matrix completion

$$f(\mathbf{X}) = \frac{1}{2} \sum_{(i,j)\in\mathcal{K}} (X_{ij} - A_{ij})^2 \qquad \mathcal{X} = \{\mathbf{X} \mid \|\mathbf{X}\|_{\text{nuc}} \le R\}$$



Dataset *MovieLens100k* https://grouplens.org/datasets/movielens/100k/ is used

Concluding remarks

- We talked about ExtraFW
  - for faster convergence using parameter-free step sizes
  - with promising performance for classification and matrix completion
- Future directions
  - More constraint-dependent accelerated rates
  - How about an adaptive manner for a local L?
- Check out our paper #1351
  `https://arxiv.org/abs/2012.05284`

THANK YOU and STAY HEALTHY!