HSE University
23 December

Step-size strategies in the conditional gradient algorithm

Ilya Levin
Pavel Zakharov
Valeriia Shcherbakova

Table of content

Frank-Wolfe algorithm

is a method for constrained optimization that solves problems of the form $\underset{\boldsymbol{x} \in \mathcal{D}}{\text{minimize}}\, f(\boldsymbol{x})$

where f is a smooth function for which we have access to its gradient and D is a compact set. We also assume to have access to a *linear minimization oracle* over D, that is, a routine that solves problems of the form

$$\boldsymbol{s}_t \in \underset{\boldsymbol{s} \in \mathcal{D}}{\arg\max} \langle -\nabla f(\boldsymbol{x}_t), \boldsymbol{s} \rangle$$

Frank-Wolfe algorithm pseudo code

Frank–Wolfe algorithm

Input: initial guess $\boldsymbol{x}_0$, tolerance $\delta > 0$

For $t = 0, 1, \ldots$ **do** $\quad \boldsymbol{s}_t \in \arg\max_{\boldsymbol{s} \in \mathcal{D}} \langle -\nabla f(\boldsymbol{x}_t), \boldsymbol{s} \rangle$

Set $\boldsymbol{d}_t = \boldsymbol{s}_t - \boldsymbol{x}_t$ and $g_t = \langle -\nabla f(\boldsymbol{x}_t), \boldsymbol{d}_t \rangle$

Choose step-size γ_t (to be discussed later)

$\boldsymbol{x}_{t+1} = \boldsymbol{x}_t + \gamma_t \boldsymbol{d}_t$.

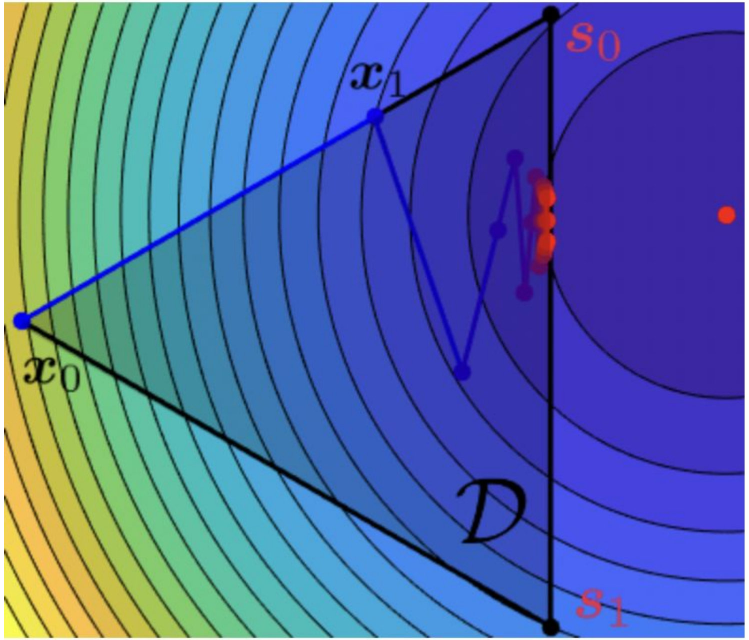
end For loop

return $\boldsymbol{x}_t$

Predefined decreasing sequence

$$\gamma_t = \frac{2}{t+2}$$

is to choose the step-size according to the pre-defined decreasing sequence



➕
1) is straightforward;
2) cheap to compute.

➖
1) in practice it performs worse than the alternatives;
2) Oscillates a lot.

Demyanov-Rubinov

$$\gamma = \min\left\{\frac{g_t}{L\|\boldsymbol{d}_t\|^2}, 1\right\}$$

is to choose the step-size according to the pre-defined decreasing sequence

\+

1) Not that expensive to compute;
2) Good convergence in practice.

\-

1) Can be problematic to estimate L;
2) Convergence in the neighbourhood of solution is not optimal.

Exact line-search

$$\gamma_\star \in \arg\min_{\gamma\in[0,1]} f(\boldsymbol{x}_t + \gamma \boldsymbol{d}_t)$$

takes the step-size that maximizes the decrease in objective along the update direction

+

1) Great convergence in practice;
2) Gives the highest decrease per iteration.

−

1) Pretty heavy computation if we don't have an access to the minimizer of f.

Backtracking line-search

$$\gamma_t = \min\left\{\frac{g_t}{M_t\|\boldsymbol{d}_t\|^2}, 1\right\}$$

is to choose the step-size according to the pre-defined decreasing sequence

$$Q_t(\gamma, M_t) \stackrel{\text{def}}{=} f(\boldsymbol{x}_t) - \gamma g_t + \frac{\gamma^2 M_t}{2}\|\boldsymbol{d}_t\|^2$$

- wildly successful and are a core part of any state-of-the-art implementation of (proximal) gradient descent and Quasi-Newton methods.

- Highly depends on the input parameters

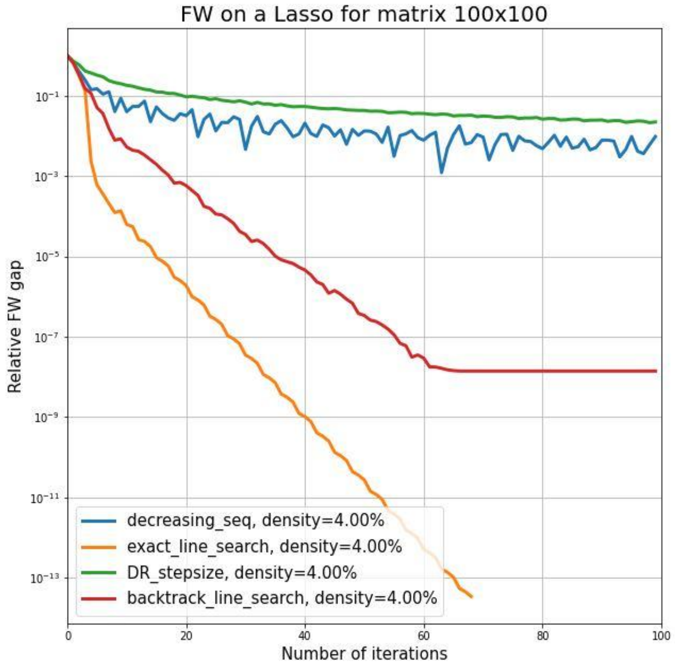
$M_t = \eta M_{t-1}$

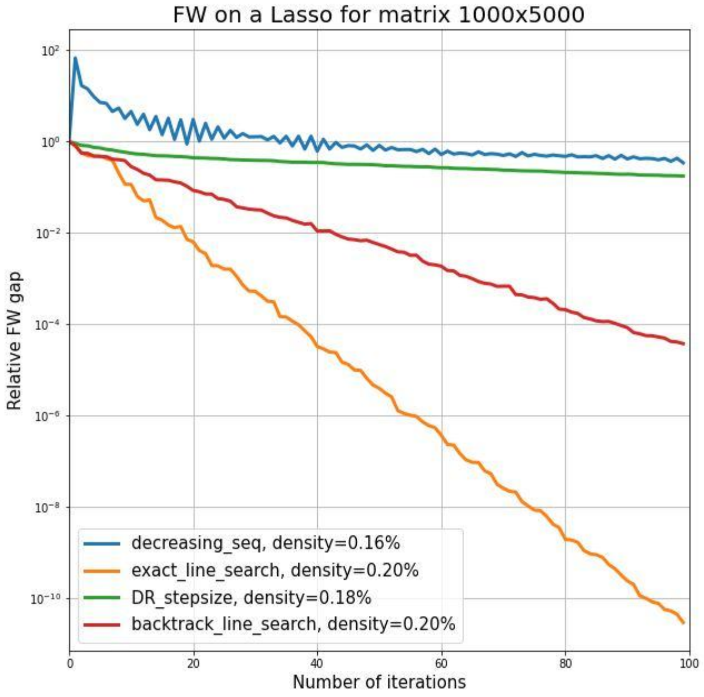
$\gamma_t = \min\{g_t/(M_t\|\boldsymbol{d}_t\|^2), 1\}$

While $f(\boldsymbol{x}_t + \gamma_t \boldsymbol{d}_t) > Q_t(\gamma_t, M_t)$ **do**

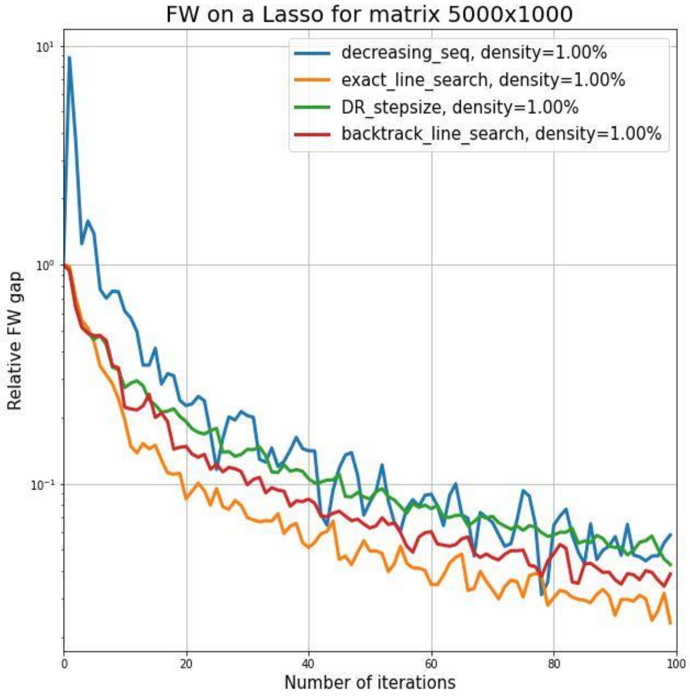
$\quad M_t = \tau M_t$

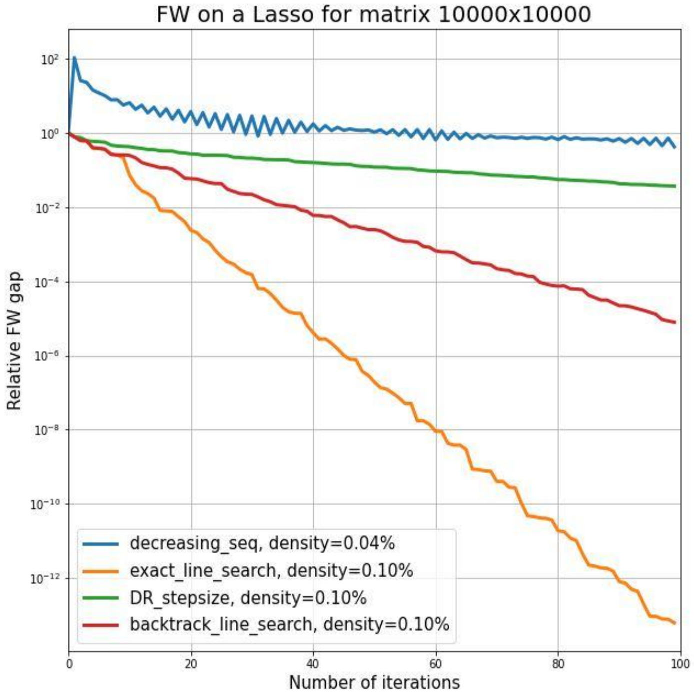
Our results: linear regression



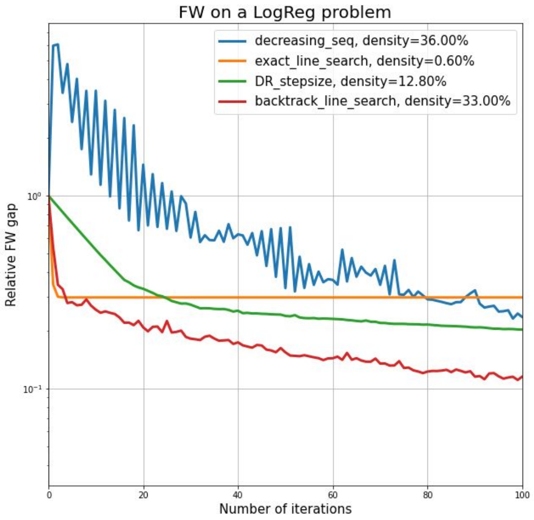


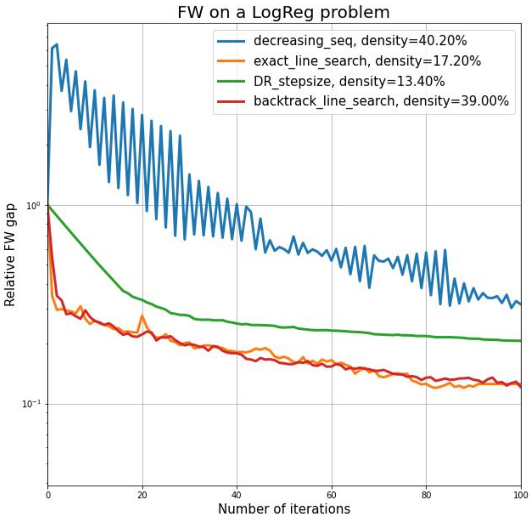
Synthetic dataset MSE:

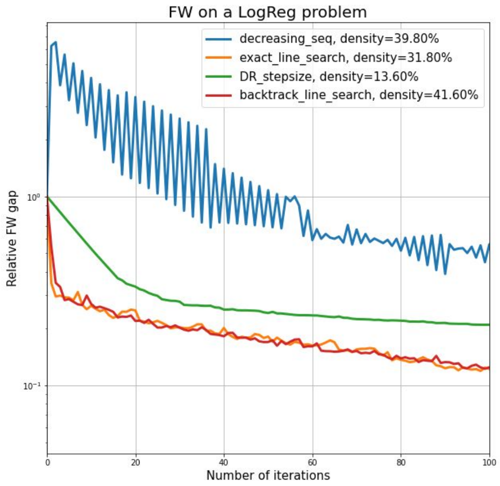




Modelon dataset logreg:







Thank you for your attention!