Iterative Double Sketching for Faster Least-Squares Optimization

Rui Wang Yanyan Ouyang Wangli Xu

Center for Applied Statistics and School of Statistics, Renmin University of China, Beijing 100872, China

Problem Setting

Overdetermined linear least-squares problem:

- Input data:
  - Data maitrx $\mathbf{A} = (\mathbf{a}_1, \ldots, \mathbf{a}_N)^\top \in \mathbb{R}^{N \times d}$ with full column rank.
  - Observations $\mathbf{y} = (y_1, \ldots, y_N)^\top \in \mathbb{R}^N$.
- Objective:

$$\mathbf{x}^* := \arg\min_{\mathbf{x} \in \mathbb{R}^d} \left\{ f(\mathbf{x}; \mathbf{A}, \mathbf{y}) := \frac{1}{2} \|\mathbf{A}\mathbf{x} - \mathbf{y}\|^2 \right\}.$$

- Exact solution:
  - Gradient $\nabla f(\mathbf{x}; \mathbf{A}, \mathbf{y}) := \mathbf{A}^\top (\mathbf{A}\mathbf{x} - \mathbf{y})$.
  - Hessian $\mathbf{A}^\top \mathbf{A}$.
  - From any initial point $\mathbf{x}_0$, one Newton iteration yields the exact solution:

$$\mathbf{x}^* = \mathbf{x}_0 - (\mathbf{A}^\top \mathbf{A})^{-1} \nabla f(\mathbf{x}_0; \mathbf{A}, \mathbf{y}).$$

  - Computing time: $O(Nd^2)$.

Iterative Hessian Sketch (IHS) algorithm

The computing time $O(Nd^2)$ can be improved if an approximate solution with small random error is allowed.

- IHS: proposed by Pilanci and Wainwright (2016).
- Idea: Sketched Hessian + Iteration
- Formula:

$$\mathbf{x}_{t+1} = \mathbf{x}_t - (\mathbf{A}^\top \mathbf{S}_t^\top \mathbf{S}_t \mathbf{A})^{-1} \nabla f(\mathbf{x}_t; \mathbf{A}, \mathbf{y}), \quad t = 1, \ldots, T,$$

  where $\mathbf{S}_0, \mathbf{S}_1, \ldots$ are i.i.d. $r \times N$ random sketching matrices.
- Computing time:
  - Lacotte and Pilanci (2020): a variant of IHS.
  - ϵ relative error: $\|\mathbf{A}(\mathbf{x}_T - \mathbf{x}^*)\| \leq \epsilon \|\mathbf{A}(\mathbf{x}_0 - \mathbf{x}^*)\|$.
  - Computing time: $O\left((\log(d) + \log(\frac{1}{\epsilon}))Nd + d^3\right)$.

The goal

Further reduce the computing time.

- Computational bottleneck of Newton method:
  - The first computational bottleneck:
    - The Hessian $\mathbf{A}^\top \mathbf{A}$.
    - Computing time $O(Nd^2)$.
  - The second computational bottleneck:
    - The gradient $\nabla f(\mathbf{x}; \mathbf{A}, \mathbf{y}) = \mathbf{A}^\top (\mathbf{A}\mathbf{x} - \mathbf{y})$.
    - Computing time $O(Nd)$.
- IHS uses sketching to approximate the Hessian.
- Our idea: approximate both the gradient and the Hessian.

Iterative Double Sketching (IDS) Framework

- Hessian sketching:
  - Fixed Hessian sketching across all iterations: $\tilde{\mathbf{S}} \in \mathbb{R}^{r \times N}$.
  - sketched Hessian: $\tilde{\mathbf{H}} := \mathbf{A}^\top \tilde{\mathbf{S}}^\top \tilde{\mathbf{S}} \mathbf{A}$.
- Initial point: $\mathbf{x}_0 := \tilde{\mathbf{H}}^{-1} \mathbf{A}^\top \tilde{\mathbf{S}}^\top \tilde{\mathbf{S}} \mathbf{y}$.
- Gradient sketching for $t = 0, \ldots, T-1$:
  - m_t: sketch size when computing $\mathbf{x}_{t+1}$.
  - $\mathbf{S}_t \in \mathbb{R}^{m_t \times N}$: gradient sketching matrix when computing $\mathbf{x}_{t+1}$.
- Update formula:

$$\mathbf{x}_{t+1} = \mathbf{x}_t - \mu \tilde{\mathbf{H}}^{-1} \nabla f(\mathbf{x}_t; \mathbf{S}_t \mathbf{A}, \mathbf{S}_t \mathbf{y}),$$

  where $\mu > 0$ is the step size parameter.
- Define $T^\dagger := \min\left(\{t : 0 \leq t < T \text{ and } m_t = N\} \cup \{T\}\right)$.
  - For $t = 0, \ldots, T^\dagger - 1$, sketched data is used to approximate the gradient;
  - For $t = T^\dagger, \ldots, T-1$, the full data is used to compute the exact gradient.

Generic IDS algorithm

Algorithm 1 Generic iterative double sketching

Input: $\mu, T, \tilde{\mathbf{S}}\mathbf{A}, (\mathbf{S}_t\mathbf{A}, \mathbf{S}_t\mathbf{y}), t = 0, \ldots, T-1$

$\tilde{\mathbf{H}}^{-1} \leftarrow (\mathbf{A}^\top \tilde{\mathbf{S}}^\top \tilde{\mathbf{S}}\mathbf{A})^{-1}$

$\mathbf{x}_0 \leftarrow \tilde{\mathbf{H}}^{-1}\mathbf{A}^\top \tilde{\mathbf{S}}^\top \tilde{\mathbf{S}}\mathbf{y}$

for $t \leftarrow 0$ **to** $T-1$ **do**

  $\mathbf{x}_{t+1} \leftarrow \mathbf{x}_t - \mu \tilde{\mathbf{H}}^{-1} \nabla f(\mathbf{x}_t; \mathbf{S}_t\mathbf{A}, \mathbf{S}_t\mathbf{y})$

end for

Return $\mathbf{x}_T$

Challenges:

1. The choice of the sketch size m_t.
2. The computation of $\mathbf{S}_t\mathbf{A}, \mathbf{S}_t\mathbf{y}$ requires $O(Nd)$ time, too slow.

Optimal choice of sketche size with Gaussian sketching

- Under certain conditions, we prove that asymptotically, the optimal sketch size is

$$m_t = \min(c\sqrt{g(t,T)}, N),$$

where $g(t,T) = C(T-t-1)\frac{d^{T-t}}{r^{T-t-1}}$, $C(n) = \frac{(2n)!}{(n+1)!n!}$ and $c > 0$ is a constant.
- Note: $\frac{m_{t+1}}{m_t}$ is approximately a constant.
- Guidance for general sketching: choose $\frac{m_{t+1}}{m_t}$ to be a constant.

IDS with iteration efficient sketching: Sketching

- We choose $\frac{m_{t+1}}{m_t} = 2$.
- Idea: sequentially compute the sketched data in reverse order $(\mathbf{S}_{T^\dagger-1}\mathbf{A}, \mathbf{S}_{T^\dagger-1}\mathbf{y}), \ldots, (\mathbf{S}_0\mathbf{A}, \mathbf{S}_0\mathbf{y})$.
- First stage, $t = T^\dagger - 1, \ldots, T^\diamond$:
  - $\mathbf{S}_t := \mathbf{G}^*_{m_t,N}\mathbf{D}_N\mathbf{P}_N$.
  - $\mathbf{D}_N \in \mathbb{R}^{N\times N}$: a diagonal matrix whose diagonal elements are i.i.d. Rademacher random variables.
  - $\mathbf{P}_N \in \mathbb{R}^{N\times N}$: a uniformly distributed permutation matrix.
  - $\mathbf{G}^*_{m,N} := \mathbf{I}_m \otimes \mathbf{1}^\top_{\frac{N}{m}}$,
- Second stage, $t = T^\diamond - 1, \ldots, 0$
  - $\mathbf{W}_{m_{T^\diamond}} \in \mathbb{R}^{m_{T^\diamond}\times m_{T^\diamond}}$: the Walsh-Hadamard transform.
  - $\mathbf{S}_t := \mathbf{G}^*_{m_t,m_{T^\diamond}}\mathbf{D}_{m_{T^\diamond}}\mathbf{P}_{m_{T^\diamond}}\mathbf{W}_{m_{T^\diamond}}\mathbf{S}_{T^\diamond}$.
- $\mathbf{S}_t\mathbf{A}$ can be computed from $\mathbf{S}_{t+1}\mathbf{A}$.
- The sketching so defined guarantees good subspace embedding property.

IDS with iteration efficient sketching: Algorithm

Algorithm 3 IDS algorithm with iteration efficient sketching matrices

Input: $\mathbf{A} \in \mathbb{R}^{N \times d}, \mathbf{y} \in \mathbb{R}^N, r, m_0, T^\diamond, T, \mu$

$T^\dagger \leftarrow \log_2(\frac{N}{m_0})$

$\mathbf{A} \leftarrow \mathbf{D}_N \mathbf{P}_N \mathbf{A}$; $\mathbf{y} \leftarrow \mathbf{D}_N \mathbf{P}_N \mathbf{y}$; $\mathbf{S}_{T^\dagger} \leftarrow \mathbf{I}_N$;

for $t \leftarrow T^\dagger - 1$ **to** 0 **do**

  $\mathbf{S}_t \mathbf{A} \leftarrow (\mathbf{I}_{2^t m_0} \otimes \mathbf{1}_2^\top) \mathbf{S}_{t+1} \mathbf{A}$; $\mathbf{S}_t \mathbf{y} \leftarrow (\mathbf{I}_{2^t m_0} \otimes \mathbf{1}_2^\top) \mathbf{S}_{t+1} \mathbf{y}$

  if $t = T^\diamond$ **then**

    $\mathbf{S}_{T^\diamond} \mathbf{A} \leftarrow \mathbf{D}_{m_{T^\diamond}} \mathbf{P}_{m_{T^\diamond}} \mathbf{W}_{m_{T^\diamond}} \mathbf{S}_{T^\diamond} \mathbf{A}$

    $\mathbf{S}_{T^\diamond} \mathbf{y} \leftarrow \mathbf{D}_{m_{T^\diamond}} \mathbf{P}_{m_{T^\diamond}} \mathbf{W}_{m_{T^\diamond}} \mathbf{S}_{T^\diamond} \mathbf{y}$

  end if

end for

$\tilde{\mathbf{H}}^{-1} \leftarrow (\mathbf{A}^\top \mathbf{S}_0^\top \mathbf{S}^{\dagger\top} \mathbf{S}^\dagger \mathbf{S}_0 \mathbf{A})^{-1}$

$\mathbf{x}_0 \leftarrow \tilde{\mathbf{H}}^{-1} \mathbf{A}^\top \mathbf{S}_0^\top \mathbf{S}^{\dagger\top} \mathbf{S}^\dagger \mathbf{S}_0 \mathbf{y}$

for $t \leftarrow 0$ **to** $T^\dagger - 1$ **do**

  $\mathbf{x}_{t+1} \leftarrow \mathbf{x}_t - \mu \tilde{\mathbf{H}}^{-1} \nabla f(\mathbf{x}_t; \mathbf{S}_t \mathbf{A}, \mathbf{S}_t \mathbf{y})$

end for

for $t \leftarrow T^\dagger$ **to** $T - 1$ **do**

  $\mathbf{x}_{t+1} \leftarrow \mathbf{x}_t - \mu \tilde{\mathbf{H}}^{-1} \nabla f(\mathbf{x}_t; \mathbf{A}, \mathbf{y})$

end for

Return $\mathbf{x}_T$

IDS with iteration efficient sketching: Theory

- Under certain conditions, with probability at least $1-3\delta$,

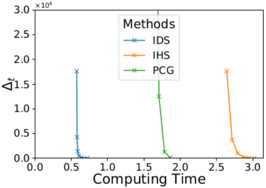
$$\|\mathbf{A}(\mathbf{x}_T-\mathbf{x}^*)\| \leq \frac{1}{2^T}\|\mathbf{A}(\mathbf{x}_0-\mathbf{x}^*)\| + \frac{4\sqrt{5}(\sqrt{2}+1)}{2^{T-T^\dagger}\delta}\sqrt{\frac{d}{N}}\|\mathbf{A}\mathbf{x}^*-\mathbf{y}\|.$$

- The computing time to achieve ϵ relative error, i.e., $\|\mathbf{A}(\mathbf{x}_T-\mathbf{x}^*)\| \leq \epsilon\|\mathbf{A}(\mathbf{x}_0-\mathbf{x}^*)\|$. Assume that $N=\Omega(d^2)$, and $\|\mathbf{A}(\mathbf{x}_0-\mathbf{x}^*)\| \asymp \sqrt{\frac{d}{r}}\|\mathbf{A}\mathbf{x}^*-\mathbf{y}\|$.

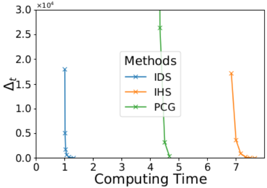
Methods	Computing time
IHS in Lacotte and Pilanci (2020)	$O\left((\log(d)+\log(\frac{1}{\epsilon}))Nd+d^3\right)$
PCG in Lacotte and Pilanci (2021)	$O\left(\left(\log(d)+\max\left(\sqrt{\log(\frac{1}{\epsilon})}, \frac{\log(\frac{1}{\epsilon})}{\log(\frac{N}{d^2})}\right)\right)Nd\right)$
IDS	$O\left(\max\left(1, \log_2(\frac{1}{\epsilon})-\frac{1}{2}\log_2(\frac{N}{d(\log(d))^3})\right)Nd+d^3\log(d)\right)$

Experiments

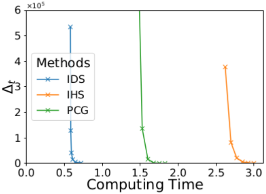
- $T^{\dagger} = 5$, $T^{\diamond} = 1$, $m_0 = N/2^5$, $r = 8d$. Following the result of Özaslan et al. (2019), we adopt the step size $\mu = \frac{(1-d/r)^2}{1+d/r}$.
- We use $\Delta_t := \|\mathbf{A}(\mathbf{x}_t - \mathbf{x}^*)\|^2$ to measure the precision of $\mathbf{x}_t$.



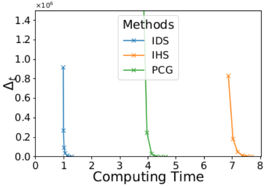
(a) Model I. $N = 2^{20}, d = 2^6$.



(b) Model I. $N = 2^{20}, d = 2^7$.



(c) Model II. $N = 2^{20}, d = 2^6$.



(d) Model II. $N = 2^{20}, d = 2^7$.

References

Lacotte, J. and Pilanci, M. (2020). Optimal randomized first-order methods for least-squares problems. In *ICML*, pages 5587–5597.

Lacotte, J. and Pilanci, M. (2021). Faster least squares optimization. arXiv:1911.02675.

Özaslan, I. K., Pilanci, M., and Arikan, O. (2019). Iterative hessian sketch with momentum. In *ICASSP*, pages 7470–7474.

Pilanci, M. and Wainwright, M. J. (2016). Iterative Hessian sketch: fast and accurate solution approximation for constrained least-squares. *Journal of Machine Learning Research*, 17:Paper No. 53, 38.