Accelerated Stochastic Matrix Inversion:

General Theory and Speeding up BFGS Rules for Faster Second-Order Optimization

Robert M. Gower[1] Filip Hanzely[2] Peter Richtárik[2, 3, 4] Sebastian U. Stich[5]

[1]Télécom ParisTech [2]KAUST [3]University of Edinburgh [4]MIPT [5]EPFL

Linear Systems in Euclidean Space

Consider the linear system

$$\mathcal{A}x = b, \quad (1)$$

where $\mathcal{A} : \mathcal{X} \mapsto \mathcal{Y}$ a linear operator, and $\mathcal{X}$ and $\mathcal{Y}$ are finite dimensional Euclidean spaces.

Optimization Problem: For $x_0 \in \mathcal{X}$, find

$$x_* \overset{\text{def}}{=} \arg\min_{x \in \mathcal{X}} \tfrac{1}{2}\|x - x_0\|^2 \quad \text{subject to} \quad \mathcal{A}x = b.$$

Motivation: Matrix Inversion

Given a symmetric pos. definite matrix $A \in \mathbb{R}^{n\times n}$,

$$A^{-1} = \arg\min_{X \in \mathbb{R}^{n\times n}} \|X\|^2_{F(A)} \overset{\text{def}}{=} \|A^{1/2}XA^{1/2}\|_F$$
$$\text{subject to } AX = I,\ X = X^\top$$

Adaptive sketch-and-project methods [1] are competitive with the state of the art.

Sketch-and-Project Methods

Now consider (1) in $\mathcal{X} = \mathbb{R}^n$ and $\mathcal{Y} = \mathbb{R}^m$. The sketch and project iteration solves the problem:

$$x_{k+1} = \text{argmin}_{x\in\mathbb{R}^n} \|x_k - x\|^2_B$$
$$\text{subject to } S_k^\top Ax = S_k^\top b,$$

where $\|x\|^2_B = \langle Bx, x\rangle$ for some $B \succ 0$ and S_k is a random sketching matrix sampled from some distribution $\mathcal{D}$.

Theorem [2, 3]

The random iterates of the sketch-and-project method converge to x_* linearly with the rate

$$\mathbf{E}\left[\|x_k - x_*\|^2_B\right] \le (1-\mu)^k \|x_0 - x_*\|^2_B,$$
$$\mu \overset{\text{def}}{=} \lambda^+_{\min}\left(\mathbf{E}\left[B^{-\frac{1}{2}}A^\top S_k (S_k^\top A B^{-1} A^\top S_k)^\dagger S_k^\top A B^{-\frac{1}{2}}\right]\right)$$

Main Contributions

- Extending [4], we analyze accelerated sketch-and-project algorithms in Euclidean spaces for solving (1). Applying these results to matrix inversion, we obtain faster stochastic algorithms for matrix inversion.
- After 48 years of research on quasi-Newton update formulas, we obtain the first accelerated quasi-Newton matrix inversion update rules! We apply these rules to optimization → faster quasi-Newton methods.

Algorithm

Algorithm 1 Accelerated Sketch-and-Project

1: **Parameters:** $\mu, \nu > 0$; $\mathcal{D}$ = distribution over random linear operators $\mathcal{S}$; choose $x_0 = v_0 \in \mathcal{X}$
2: Set $\beta = 1 - \sqrt{\frac{\mu}{\nu}}$, $\gamma = \sqrt{\frac{1}{\mu\nu}}$, $\alpha = \frac{1}{1+\gamma\nu}$.
3: **for** $k = 0, 1, \ldots$ **do**
4: $y_k = \alpha v_k + (1-\alpha)x_k$
5: Sample an independent copy $S_k \sim \mathcal{D}$
6: $g_k = \mathcal{A}^*\mathcal{S}_k^*(\mathcal{S}_k\mathcal{A}\mathcal{A}^*\mathcal{S}_k^*)^\dagger \mathcal{S}_k(\mathcal{A}y_k - b)$
7: $x_{k+1} = y_k - g_k$; $v_{k+1} = \beta v_k + (1-\beta)y_k - \gamma g_k$
8: **end for**

$$\mu \overset{\text{def}}{=} \inf_{x\in\mathbf{Range}(\mathcal{A}^*)} \frac{\langle \mathbf{E}[Z]x, x\rangle}{\langle x, x\rangle} \qquad \text{("strong convexity")}$$
$$\nu \overset{\text{def}}{=} \sup_{x\in\mathbf{Range}(\mathcal{A}^*)} \frac{\langle \mathbf{E}[Z\mathbf{E}[Z]^\dagger Z]x, x\rangle}{\langle \mathbf{E}[Z]x, x\rangle} \qquad \text{(new parameter)}$$
$$Z \overset{\text{def}}{=} \mathcal{A}^*\mathcal{S}_k^*(\mathcal{S}_k\mathcal{A}\mathcal{A}^*\mathcal{S}_k^*)^\dagger \mathcal{S}_k\mathcal{A}$$

Lemma

$1 \le \nu \le \frac{1}{\mu} = \|\mathbf{E}[Z]^\dagger\|$ and if $\mathbf{Range}(\mathcal{A}^*) = \mathcal{X}$, then $\frac{\mathbf{Rank}(\mathcal{A}^*)}{\mathbf{E}[\mathbf{Rank}(Z)]} \le \nu$.

Example (Linear systems in $\mathbb{R}^n$)
If $A \succ 0$, choose $B = A$ and $S = e_i$ (ith standard basis vector in $\mathbb{R}^n$) with probability proportional to A_{ii}. Then $\mu = \frac{\lambda_{\min}(A)}{\mathbf{Tr}(A)}$ and $\nu = \frac{\mathbf{Tr}(A)}{\min_i A_{ii}}$.

Main Theorem

If $\mathbf{Null}(\mathcal{A}) = \mathbf{Null}(\mathbf{E}[Z])$, then (exactness)

$$\mathbf{E}\left[\|v_k - x_*\|^2_{\mathbf{E}[Z]^\dagger} + \tfrac{1}{\mu}\|x_k - x_*\|^2\right]$$
$$\le \left(1 - \sqrt{\tfrac{\mu}{\nu}}\right)^k \mathbf{E}\left[\|v_0 - x_*\|^2_{\mathbf{E}[Z]^\dagger} + \tfrac{1}{\mu}\|x_0 - x_*\|^2\right]$$

References

[1] Robert M Gower and Peter Richtárik. Randomized quasi-Newton updates are linearly convergent matrix inversion algorithms. *SIAM Journal on Matrix Analysis and Applications*, 38(4):1380–1409, 2017.

[2] Robert Mansel Gower and Peter Richtárik. Randomized iterative methods for linear systems. *SIAM Journal on Matrix Analysis and Applications*, 36(4):1660–1690, 2015.

[3] Peter Richtárik and Martin Takáč. Stochastic reformulations of linear systems: algorithms and convergence theory. *arXiv:1706.01108*, 2017.

[4] Peter Richtárik and Martin Takáč. Stochastic reformulations of linear systems: Accelerated method. *Manuscript, October 2017*, 2017.

Accelerated BFGS Updates

Optimization Problem:

$$\min_{w\in\mathbb{R}^n} f(w),$$

for $f : \mathbb{R}^n \to \mathbb{R}$ convex and sufficiently smooth.

Quasi-Newton Methods:

$$w_{k+1} = w_k - X_k \nabla f(w_k),$$

where $X_k \approx (\nabla^2 f(w_k))^{-1}$.

Quasi-Newton update: (Secant equation)

$$X_k(\nabla f(w_k) - \nabla f(w_{k-1})) = w_k - w_{k-1}, \quad X_k = X_k^\top.$$

This can also be written as

$$X_{k+1} = \text{argmin}_X \|X - X_k\|^2_{F(A)}$$
$$\text{s.t. } X(w_{k+1} - w_k) = \nabla f(w_{k+1}) - \nabla f(w_k)$$
$$X = X^\top.$$

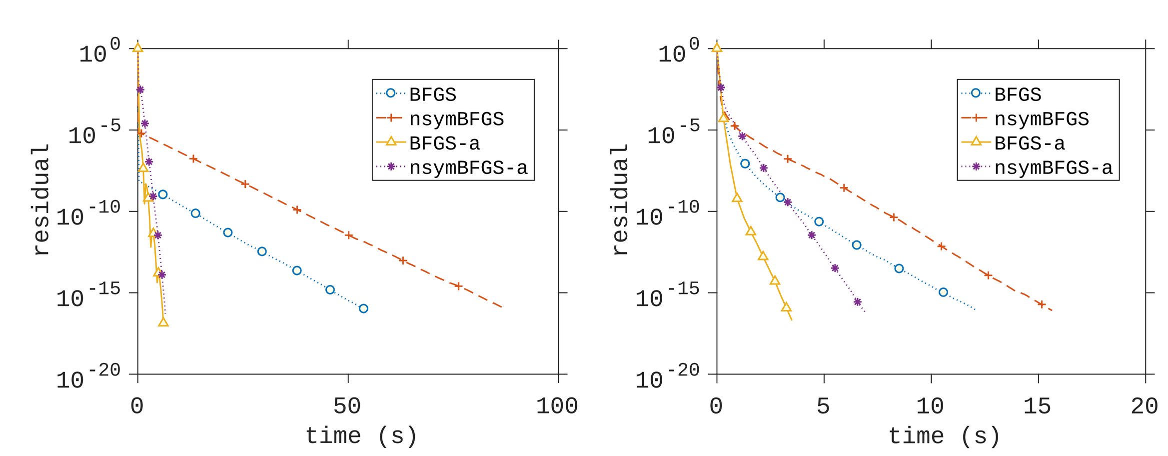
Algorithm 2 BFGS method with accelerated BFGS update

1: **Parameters:** $\mu, \nu > 0$, stepsize η.
2: Choose $X_0 \in \mathcal{X}$, w_0 and set $V_0 = X_0$, $\beta = 1 - \sqrt{\frac{\mu}{\nu}}$, $\gamma = \sqrt{\frac{1}{\mu\nu}}$, $\alpha = \frac{1}{1+\gamma\nu}$.
3: **for** $k = 0, 1, \ldots$ **do**
4: $w_{k+1} = w_k - \eta X_k \nabla f(w_k)$
5: $s_k = w_{k+1} - w_k$, $\quad \zeta_k = \nabla f(w_{k+1}) - \nabla f(w_k)$
6: $Y_k = \alpha V_k + (1-\alpha)X_k$
7: $X_{k+1} = \frac{\delta_k\delta_k^\top}{\delta_k^\top\zeta_k} + \left(I - \frac{\delta_k\zeta_k^\top}{\delta_k^\top\zeta_k}\right) Y_k \left(I - \frac{\zeta_k\delta_k^\top}{\delta_k^\top\zeta_k}\right)$
8: $V_{k+1} = \beta V_k + (1-\beta)Y_k - \gamma(Y_k - X_{k+1})$
9: **end for**

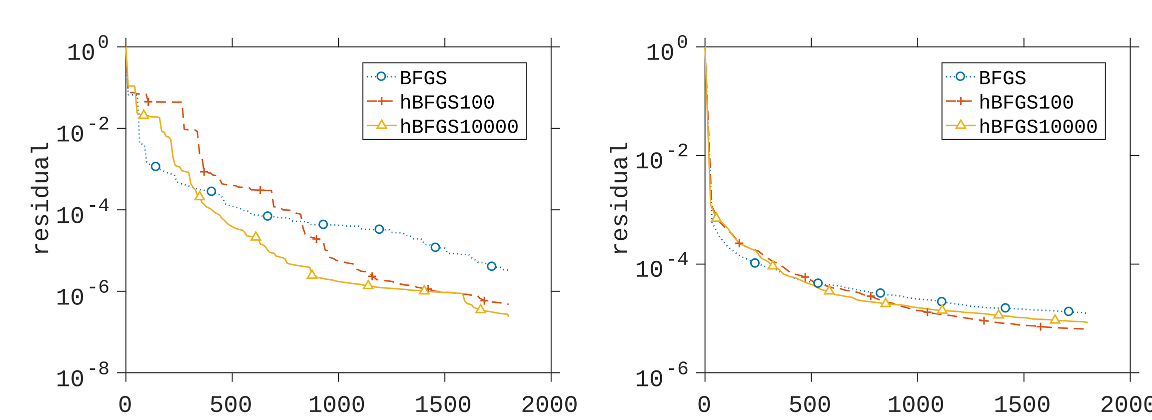
Remark: Here the Sketch-and-Project update is deterministic, the theory does not apply.

Experiments

Accelerated Matrix Inversion

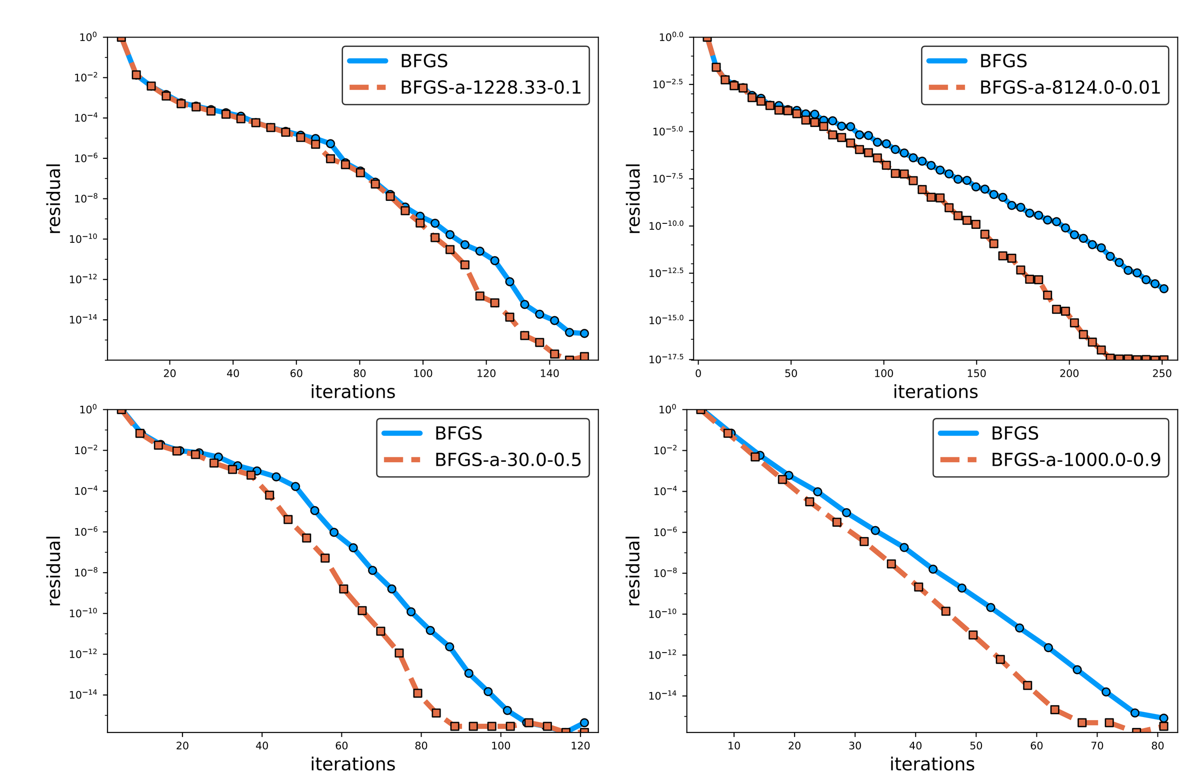


Left: Eigenvalues of $A \in \mathbb{R}^{100\times100}$ are $1, 10^3, 10^3, \ldots, 10^3$ and coordinate sketches with probabilities proportional to diag(A) are used. Right: Eigenvalues of $A \in \mathbb{R}^{100\times100}$ are $1, 2, \ldots, n$ and Gaussian sketches are used. Label "nsym" indicates non-enforcing symmetry and "-a" indicates acceleration.



Left: Epsilon dataset ($n = 2000$), uniform coordinate sketches. Right: SVHN ($n = 3072$), coordinate sketches with probabilities proportiaonal to diag(A). We choose $\mu = \frac{1}{100\nu}$ or $\mu = \frac{1}{10000\nu}$.

BFGS with accelerated update



Algorithm 2 vs standard BFGS. From left to right: `phishing`, `mushrooms`, `australian` and `splice` dataset. Acceleration parameters chosen via grid search.

Future Challenges

- Limited memory updates
- Convergence guarantees for Algorithm 2
- Optimal sketches
- Adaptive sketches