Randomized Sketching for Convex Optimization

Mert Pilanci

June 5, 2019

Core Idea

$$\arg\min_{x} \; f(Ax)$$

- A is an $n \times d$ data matrix
- sketching matrix S is an $m \times n$ matrix where $m \ll n$

$$\arg\min_{x \in \mathcal{C}} \; f(SAx)$$

- example sub-sampling $S = \begin{bmatrix} I_{m \times m} & 0_{m \times (n-m)} \end{bmatrix}$
- sub-sampling can't give any optimality guarantees!
- Randomly generate sketching matrix S, e.g., i.i.d. random

Random projections of convex programs

Original program based on data vector $y \in \mathbb{R}^n$ and data matrix $A \in \mathbb{R}^{n \times d}$:

$$x^* = \arg\min_{x \in \mathcal{C}} \underbrace{\|Ax - y\|_2^2}_{f(x)}$$

where $\mathcal{C}$ is a convex set in $\mathbb{R}^d$.

Random projections of convex programs

Original program based on data vector $y \in \mathbb{R}^n$ and data matrix $A \in \mathbb{R}^{n \times d}$:

$$x^* = \arg\min_{x \in \mathcal{C}} \underbrace{\|Ax - y\|_2^2}_{f(x)}$$

where $\mathcal{C}$ is a convex set in $\mathbb{R}^d$.

Given a sketching matrix $S \in \mathbb{R}^{m \times n}$, consider the smaller version

$$\widehat{x} = \arg\min_{x \in \mathcal{C}} \|SAx - Sy\|_2^2$$

Random projections of convex programs

Original program based on data vector $y \in \mathbb{R}^n$ and data matrix $A \in \mathbb{R}^{n \times d}$:

$$x^* = \arg\min_{x \in \mathcal{C}} \underbrace{\|Ax - y\|_2^2}_{f(x)}$$

where $\mathcal{C}$ is a convex set in $\mathbb{R}^d$.

Given a sketching matrix $S \in \mathbb{R}^{m \times n}$, consider the smaller version

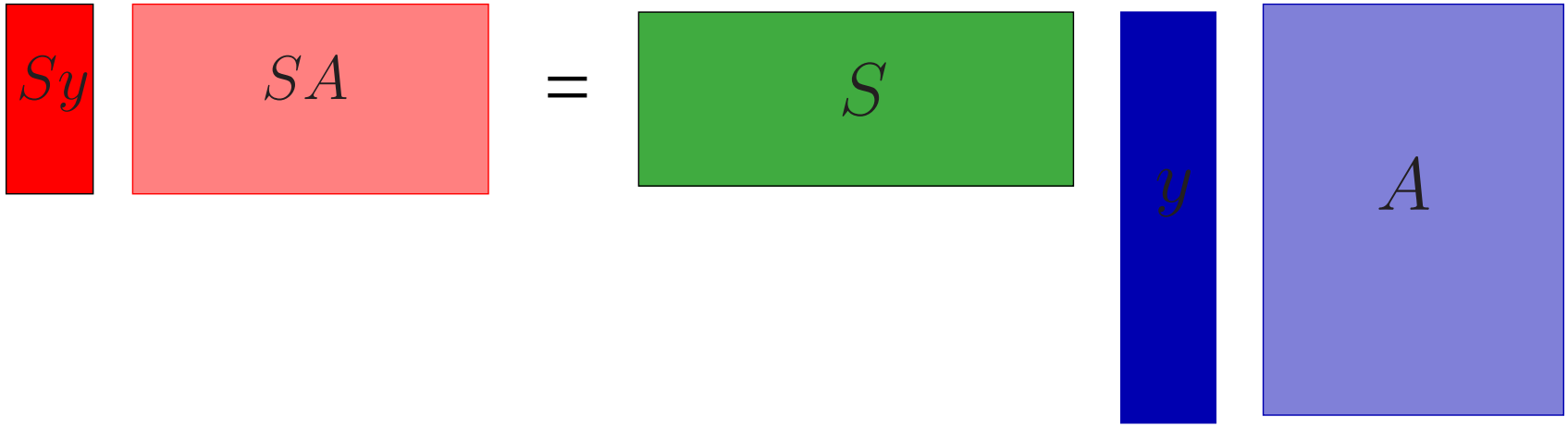
$$\widehat{x} = \arg\min_{x \in \mathcal{C}} \|SAx - Sy\|_2^2$$

Question:

How small can m be for a δ-approximation of the cost?

$$\underbrace{f(x^*)}_{\text{Optimal value}} \leq \underbrace{f(\widehat{x})}_{\text{Sketched value}} \leq \underbrace{(1+\delta)^2}_{\text{Approx. factor}} f(x^*).$$

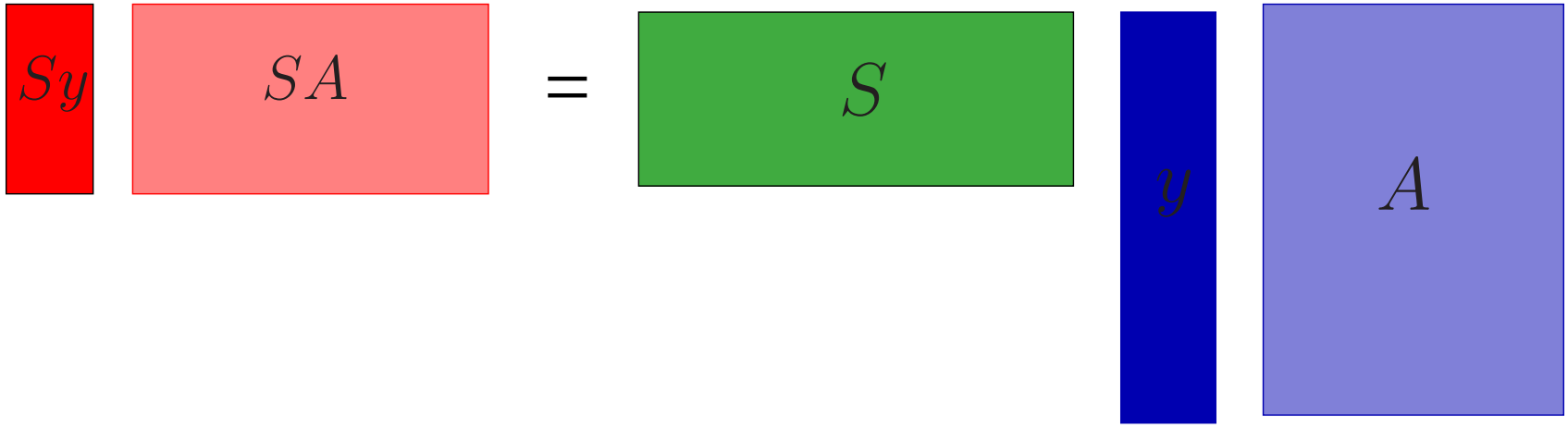
Example: Unconstrained least squares



Original problem based on data $(y, A) \in \mathbb{R}^n \times \mathbb{R}^{n \times d}$:

$$x^* = \arg\min_{x \in \mathbb{R}^d} \|Ax - y\|_2^2$$

Example: Unconstrained least squares



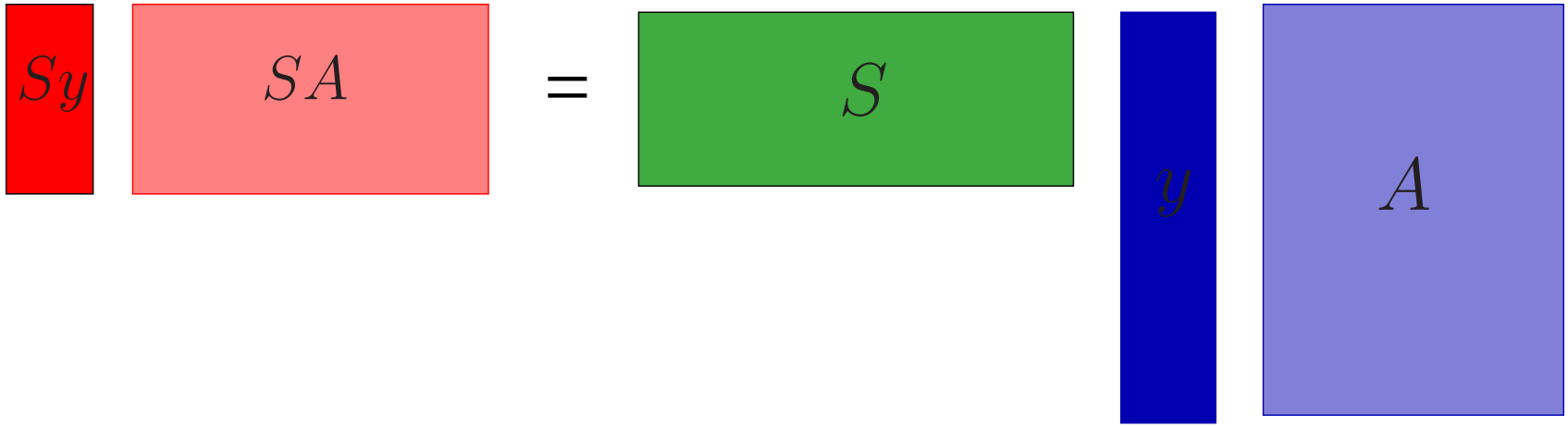
Original problem based on data $(y, A) \in \mathbb{R}^n \times \mathbb{R}^{n \times d}$:

$$x^* = \arg\min_{x \in \mathbb{R}^d} \|Ax - y\|_2^2$$

Sketched data $(Sy, SA) \in \mathbb{R}^m \times \mathbb{R}^{m \times d}$:

$$\widehat{x} = \arg\min_{x \in \mathbb{R}^d} \|SAx - Sy\|_2^2$$

Example: Unconstrained least squares



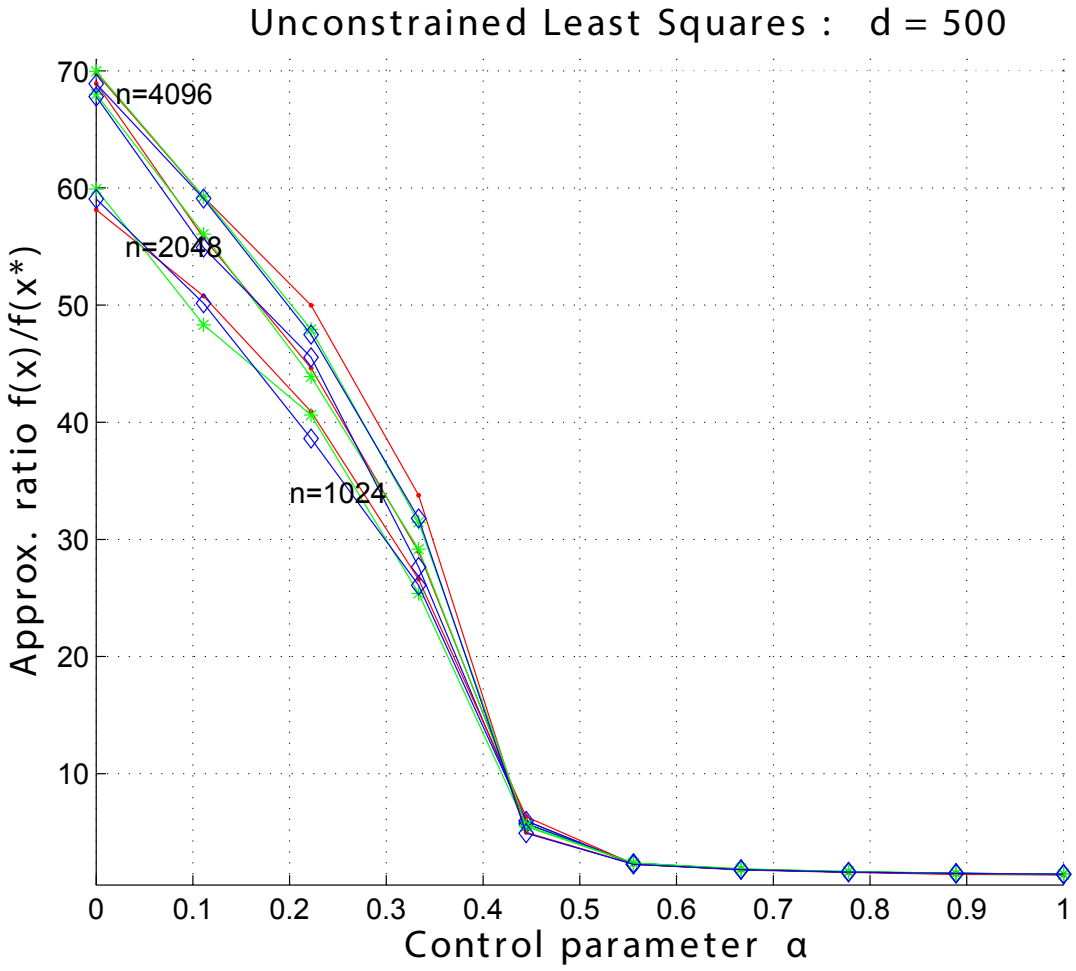
Sketched data $(Sy, SA) \in \mathbb{R}^m \times \mathbb{R}^{m \times d}$:

$$\widehat{x} = \arg\min_{x \in \mathbb{R}^d} \|SAx - Sy\|_2^2$$

Known results (Sarlós, 2006) : Let S be a Gaussian random matrix

- Take $m \gtrsim \frac{1}{\delta^2} d$, then $\widehat{x}$ is δ-optimal.
  e.g., $\delta = 0.5$ and $m = 4d$.

Empirical performance for unconstrained LS



Sketch size $m = 2\,\alpha\ \mathrm{rank}(A)$

Example: Support vector machines

- given labeled pairs $(b_i, z_i) \in \mathbb{R}^d \times \{-1, +1\}$, find linear classifier via

$$w^* = \arg\min_{w \in \mathbb{R}^d} \sum_{i=1}^{n} \underbrace{\ell(w, b_i, z_i)}_{\text{Hinge loss}^2 = \max\{0,\, 1 - z_i \langle w,\, b_i \rangle\}^2} + \frac{1}{2}\|w\|_2^2$$

Example: Support vector machines

- given labeled pairs $(b_i, z_i) \in \mathbb{R}^d \times \{-1, +1\}$, find linear classifier via

$$w^* = \arg\min_{w \in \mathbb{R}^d} \quad \sum_{i=1}^{n} \underbrace{\ell(w, b_i, z_i)}_{\text{Hinge loss}^2 = \max\{0,\, 1 - z_i \langle w,\, b_i \rangle\}^2} + \frac{1}{2}\|w\|_2^2$$

- Dual problem: $A = [b_1, ..., b_n] diag(z)$

$$x^* = \arg\min_{x \in \mathbb{R}^n_+ : 1^T x = 1} \|Ax\|_2^2$$

How to generalize to arbitrary functions ?

Consider minimizing a convex objective, where $A \in \mathbb{R}^{n \times d}$ is a data matrix

$$x^* = \arg\min_{x \in \mathcal{C}} \; g(Ax)$$

and $\mathcal{C}$ is a convex set

Introducing Newton Sketch

- Newton's Method

$$x^{t+1} = \arg\min_{x \in \mathcal{C}} \ \langle \nabla f(x^t),\, x - x^t \rangle + \frac{1}{2} \| \nabla^2 f(x^t)^{1/2} (x - x^t) \|_2^2$$

Introducing Newton Sketch

- Newton's Method

$$x^{t+1} = \arg\min_{x \in \mathcal{C}} \ \langle \nabla f(x^t),\, x - x^t \rangle + \frac{1}{2} \| \nabla^2 f(x^t)^{1/2} (x - x^t) \|_2^2$$

Definition (Newton Sketch)

$$x^{t+1} = \arg\min_{x \in \mathcal{C}} \ \langle \nabla f(x^t),\, x - x^t \rangle + \frac{1}{2} \| S^t \nabla^2 f(x^t)^{1/2} (x - x^t) \|_2^2$$

converges for self-concordant functions

Application : Linear Programming

Consider the following LP in standard form where $A \in \mathbb{R}^{n \times d}$

$$\min_{Ax \leq b} \langle c, x \rangle$$

The standard practice: interior-point methods using log-barrier

$$\min_x \underbrace{c^T x - \mu \sum_{i=1}^{n} \log(b_i - a_i^T x)}_{f(x)}$$

Hessian of $f(x)$

$$\nabla^2 f(x) = A^T diag\left(\frac{1}{(b_i - a_i^T x)^2}\right) A$$

takes $O(nd^2)$ operations to compute exactly.

Application : Linear Programming

- Hessian of $f(x) = c^T x - \sum_{i=1}^{n} \log(b_i - a_i^T x)$

$$\nabla^2 f(x) = A^T diag\left(\frac{1}{(b_i - a_i^T x)^2}\right) A \ ,$$

Application : Linear Programming

- Hessian of $f(x) = c^T x - \sum_{i=1}^{n} \log(b_i - a_i^T x)$

$$\nabla^2 f(x) = A^T diag\left(\frac{1}{(b_i - a_i^T x)^2}\right) A \ ,$$

- Root of the Hessian

$$(\nabla^2 f(x))^{1/2} = diag\left(\frac{1}{|b_i - a_i^T x|}\right) A \ ,$$

Application : Linear Programming

- Hessian of $f(x) = c^T x - \sum_{i=1}^{n} \log(b_i - a_i^T x)$

$$\nabla^2 f(x) = A^T diag\left(\frac{1}{(b_i - a_i^T x)^2}\right) A \ ,$$

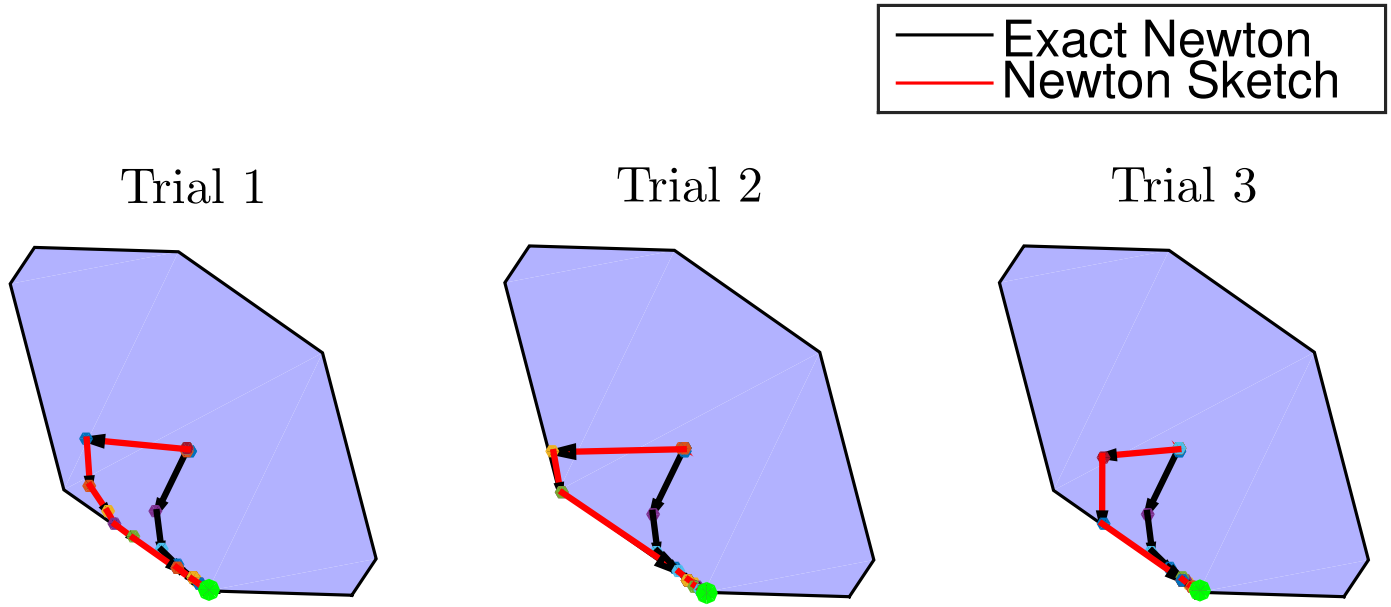
- Root of the Hessian

$$(\nabla^2 f(x))^{1/2} = diag\left(\frac{1}{|b_i - a_i^T x|}\right) A \ ,$$

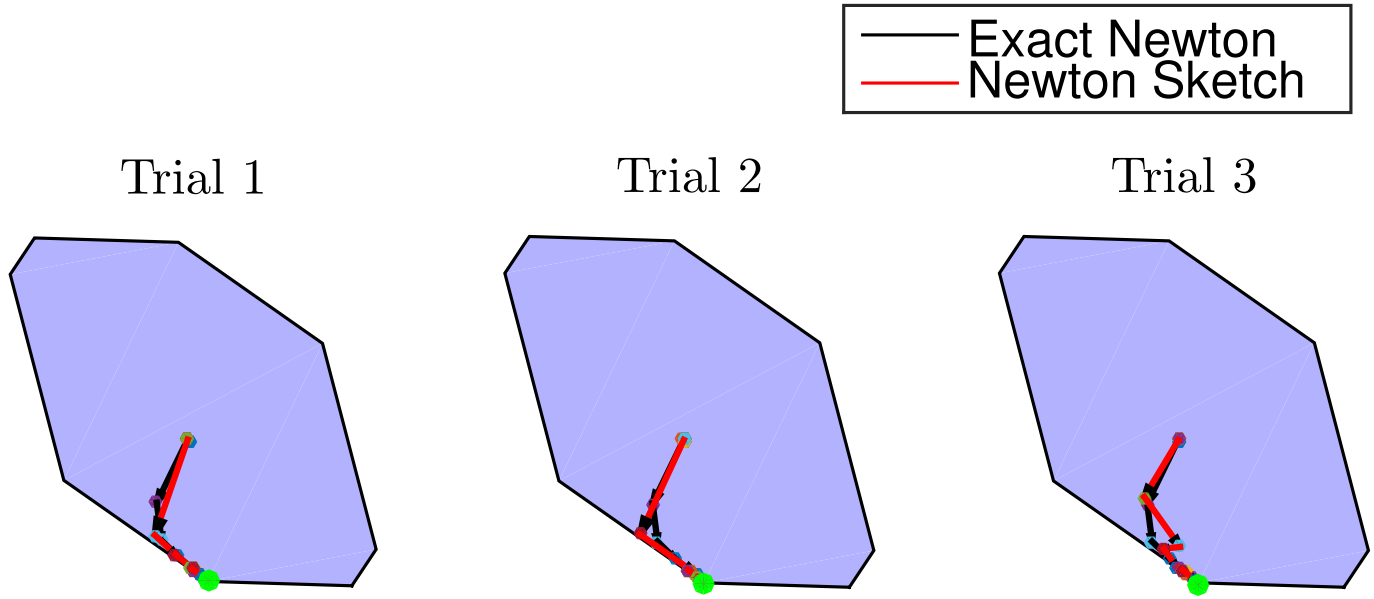
- Sketch of the Hessian

$$S^t(\nabla^2 f(x))^{1/2} = S^t diag\left(\frac{1}{|b_i - a_i^T x|}\right) A$$

takes $O(md^2)$ operations



(a) sketch size $m = d$



(b) sketch size $m = 4d$

Linear Programming optimality gap vs CPU-time (n=10000, d=100)