Matrix Sketching: the Johnson-Linderstrauss Lemma.

Thomas Moreau INRIA Saclay

Approximate solution of least square

We want to solve approximately the linear system

$$y = X\beta + \epsilon$$

where $y \in \mathbb{R}^n$, $X \in \mathbb{R}^{n\times p}$ with $1 \ll p \ll n$

Ordinary Least Square: $\beta_{OLS} = \arg\min_\beta \|y - X\beta\|_2^2$

Sketching: Choose $S \in \mathbb{R}^{m\times n}$ with $m \ll n$ and solve

$$\beta_S = \arg\min_\beta \|Sy - SX\beta\|_2^2$$

Question: How to ensure efficient solution with good precision?

Small m with $\|y - X\beta_S\|_2^2 \leq (1+\epsilon)\|y - X\beta_{OLS}\|_2^2$

Theorem - Johnson & Linderstrauss (1984)

Given $0 < \epsilon < 1$ and for n points $\{x_1, \dots, x_n\}$, there is a linear embedding $f : \mathbb{R}^p \to \mathbb{R}^m$ with $m = \mathcal{O}(\frac{\log(n)}{\epsilon^2})$ *s.t.*

$$(1-\epsilon)\|x_i - x_j\|_2 \leq \|f(x_i) - f(x_j)\|_2 \leq (1+\epsilon)\|x_i - x_j\|_2$$

Summary:

- One can map n vectors to $\mathcal{O}(\log(n))$ dim while preserving Euclidean geometry.
- The scaling m is optimal.

How to make scketching fast

Issue: naive linear mapping is dense $\rightarrow$ hard to store/compute.

Fondamental issue: For a sparse vector, unless you get all coordinates, you have a high probability to map it to 0.

Solution: Find structured projection such that you can have fast transforms with well spread information.

References:

- Blog post: the Johnson-Linderstrauss Lemma by Afonso Bandeira.
- Monograph: *Randomized algorithms for matrices and data* by Michael Mahoney.
- NeurIPS 2020 tutorial: *Sketching and Streaming Algorithms* by Jelani Nelson