Dimensionality Reduction for Tukey Regression

Kenneth L. Clarkson[1] **Ruosong Wang**[2] David P. Woodruff[2]

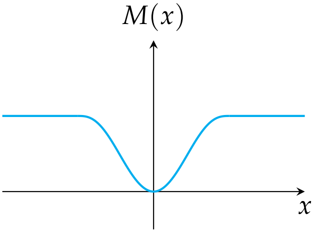
[1]IBM Research - Almaden

[2]Carnegie Mellon University

Motivation

- A number of problems in numerical linear algebra have witnessed remarkable speedups via linear sketching.
- For linear regression, we have $\mathrm{nnz}(A) + \mathrm{poly}(d/\varepsilon)$ time algorithms for a variety of convex loss functions.
- Can we apply the technique of linear sketching to non-convex loss functions, e.g., the Tukey loss function?

$$M(x) = \begin{cases} x^2 & |x| \leq 1 \\ 1 & |x| > 1 \end{cases}$$



Row Sampling Algorithm

- **Theorem 1** For a matrix $A \in \mathbb{R}^{n \times d}$ and $b \in \mathbb{R}^n$, there is a row sampling algorithm that returns a weight vector $w \in \mathbb{R}^n$, such that for

$$\hat{x} = \operatorname{argmin} \sum_{i=1}^{n} w_i M((Ax - b)_i),$$

we have

$$\sum_{i=1}^{n} M((A\hat{x} - b)_i) \leq (1 + \varepsilon) \min \sum_{i=1}^{n} M((Ax - b)_i).$$

The weight vector w has at most $\mathrm{poly}(d \log n / \varepsilon)$ non-zero entries and can be computed in $\widetilde{O}(\mathrm{nnz}(A) + \mathrm{poly}(d \log n / \varepsilon))$ time.

Oblivious Sketch

- **Theorem 2** There is a distribution $S \in \mathbb{R}^{\mathrm{poly}(d \log n) \times n}$ over sketching matrices and weight vector $w \in \mathbb{R}^n$, such that for

$$\hat{x} = \operatorname{argmin} \sum_{i=1}^{n} w_i M((SAx - Sb)_i),$$

we have

$$\sum_{i=1}^{n} M((A\hat{x} - b)_i) \le O(\log n) \min \sum_{i=1}^{n} M((Ax - b)_i).$$

- Calculating SA and Sb requires $\mathrm{nnz}(A)$ time.
- The sketch can be readily implemented in streaming and distributed settings.

Technical Lemma

- Structural Lemma for Tukey Loss Function
  - **Lemma 1** For a given matrix $A \in \mathbb{R}^{n \times d}$, there is a set of indices $I \subseteq [n]$ with size $|I| \leq \mathrm{poly}(d\alpha)$, such that for any $y = Ax$ with $\sum_{i=1}^{n} M(y_i) \leq \alpha$, for all $i \in [n]$ with $|y_i| \geq 1$, we have $i \in I$.
  - The set I can be efficiently constructed.
- Net Argument For Tukey Loss Function

For more details, hardness results, provable algorithms and experiments, please come to poster #208!