Lecture 5-1 — 10/05/2017

Prof. David Woodruff *Scribe: Hongyang Zhang*

1 Constructing a Coreset

Let $A = U\Sigma V^T$ be the SVD of A. Let $m = k + k/\epsilon$. Let Σ_m agree with Σ on the first m diagonal entries, and be 0 otherwise. We have the following claim.

Claim 1. For all projection matrices $Y = I - X$ onto $(d-k)$-dimensional subspaces, we have

$$\|\Sigma_m V^T Y\|_F^2 + c = (1 \pm \epsilon)\|AY\|_F^2, \tag{1}$$

where $c = \|A - A_m\|_F^2$ does not depend on Y and A_m is the best rank-m approximation of A.

Remark 1. We can think of S as U_m^T so that $SA = U_m^T U\Sigma V^T = \Sigma_m V^T$ is a sketch.

Proof. We note that

$$\begin{aligned} \|AY\|_F^2 &= \|U\Sigma_m V^T Y\|_F^2 + \|U(\Sigma - \Sigma_m)V^T Y\|_F^2 \\ &\le \|\Sigma_m V^T Y\|_F^2 + \|A - A_m\|_F^2 \\ &= \|\Sigma_m V^T Y\|_F^2 + c. \end{aligned} \tag{2}$$

Also

$$\begin{aligned} &\|\Sigma_m V^T Y\|_F^2 + \|A - A_m\|_F^2 - \|AY\|_F^2 \\ &= \|\Sigma_m V^T\|_F^2 - \|\Sigma_m V^T X\|_F^2 + \|A - A_m\|_F^2 - \|A\|_F^2 + \|AX\|_F^2 \\ &= \|AX\|_F^2 - \|\Sigma_m V^T X\|_F^2 \\ &= \|(\Sigma - \Sigma_m)V^T X\|_F^2 \\ &\le \|(\Sigma - \Sigma_m)V^T\|^2 \|X\|_F^2 \\ &\le \sigma_{m+1}^2 k \\ &\le \epsilon\sigma_{m+1}^2 (m - k) \\ &\le \epsilon \sum_{i \in \{k+1,\ldots,m+1\}} \sigma_i^2 \\ &\le \epsilon\|A - A_k\|_F^2 \\ &\le \epsilon\|AY\|_F^2, \end{aligned} \tag{3}$$

as desired. ∎

We can thus apply Claim 1 to construct a coreset. Suppose we have matrices $A^1, ..., A^s$ and construct $\Sigma_m^1 V^{T,1}, \Sigma_m^2 V^{T,2}, ..., \Sigma_m^s V^{T,s}$, together with $c_1, ..., c_s$. Then

$$\sum_i \|\Sigma_m^i V^{T,i} Y\|_F^2 + c_i = (1 \pm \epsilon)\|AY\|_F^2, \tag{4}$$

where A is the matrix formed by concatenating the rows of $A^1, ..., A^s$. Let B be the matrix obtained by concatenating the rows of $\Sigma_m^1 V^{T,1}, \Sigma_m^2 V^{T,2}, ..., \Sigma_m^s V^{T,s}$. Suppose we compute $B = U\Sigma V^T$ and compute $\Sigma_m V^T$ and $\|B - B_m\|_F^2$. Then

$$\|\Sigma_m V^T Y\|_F^2 + c + \sum_i c_i = (1 \pm \epsilon)\|BY\|_F^2 + \sum_i c_i = (1 \pm O(\epsilon))\|AY\|_F^2. \tag{5}$$

So $\Sigma_m V^T$ and the constant $c + \sum_i c_i$ are a coreset for A.

2 [FSS] Row-Partition Protocol

Based on the construction of coreset, the row-partition protocol is as follows:

- Server t sends the top $k/\epsilon + k$ principal components of P^t, scaled by the top $k/\epsilon + k$ singular values Σ^t, together with c^t;
- Coordinator returns top k principal components of $[\Sigma^1 V^1; \Sigma^2 V^2; ...; \Sigma^s V^s]$.

However, there are several problems for the row-partition protocol.

- sdk/ϵ real numbers of communication
- bit complexity can be large
- running time for SVDs
- does not work in arbitrary partition model

This is an SVD-based protocol. Maybe our random matrix techniques can improve communication just like they improved computation? The [KVW] protocol in the following section will handle problems 2, 3, and 4.

3 [KVW] Arbitrary Partition Model Protocol

In the arbitrary partition model, the customer-product matrix is $A = A^1 + A^2 + ... + A^s$ with arbitrary partition. Arbitrary partition model protocol is inspired by the sketching algorithm. Let S be one of the $k/\epsilon \times n$ random matrices, e.g., Gaussian sketch, CountSketch, etc. We note that S can be generated pseudorandomly from small seed. Coordinator can also send small seed for S to all servers. We can do the followings: Server t computes SA^t and sends it to coordinator. The coordinator sends $\sum_{t=1}^s SA^t = SA$ to all servers. There is a good k-dimensional subspace inside of SA. If we knew it, t-th server could output projection of A^t onto it.

However, there are some problems for this protocol:

- Cannot output projection of A^t onto SA since the rank is too large;

- Could communicate this projection to the coordinator who could find a k-dimensional space, but communication depends on n.

To fix this, instead of projecting A onto SA, we can solve

$$\min_{\text{rank}-kX} \|A(SA)^T XSA - A\|_F^2. \tag{6}$$

Let T_1 and T_2 be affine embeddings, we can quickly solve

$$\min_{\text{rank}-kX} \|T_2 A(SA)^T XSAT_2 - T_1 AT_2\|_F^2. \tag{7}$$

This is because the optimization problem is small and has a closed-form solution. Everyone can then compute XSA and then output k directions.