Learning-Based Low-Rank Approximations

Piotr Indyk        Ali Vakilian        Yang Yuan
Learning-Based Low Rank Approx.

**Low Rank Approx.**
Find a rank-k $A'$, s.t.
$$||A-A'||_F \leq (1+\epsilon) ||A-A_k||_F$$
More efficient than computing $A_k$
- Sarlos’06
- Clarkson-Woodruff’09,13

*Refer to them as SCW algorithm*

**Main Approach:** linear sketches
- Perform SVD on $RA$
- $R$ can be dense or sparse
- $R$ is usually a random matrix

**Our Technique (Learned Sketch)**
Sample matrices $A_1...A_N$
Find $S$ that minimizes total loss, i.e.,
$$\sum_i ||A_i - SCW(S, A_i)||_F$$

**Details:**
- Use sparse matrices $S$
- Optimize using SGD in Pytorch
- Need to differentiate the above w.r.t. $S$
- Represent SVD as a sequence of power-method applications (each is differentiable)

**Sketch Monotonicity:** augmenting $R + learned sketch$ $S$ cannot increase the total loss
Our algorithm with $(R + S)$ inherits worst-case guarantees from $R$
Sarlos-ClarksonWoodruff Framework

**Streaming algorithm (two passes)**
- Compute SA (first pass)
- Compute orthonormal V that spans rowspace of SA
- Compute $AV^T$ (second pass)
- Return $SCW(S,A):= [AV^T]_k V$

**Space Complexity**
- Suppose that A is $n \times d$, S is $m \times n$
- Then SA is $m \times d$, $AV^T$ is $n \times m$
- Space proportional to $m$
- Theory: $m = O(k/\varepsilon)$

**Classic Approach:**
- S: sparse random matrix
- Worst-case theoretical bounds

**Our Learning-Based Approach**
- S: *learned matrix* from training
- Better empirical performance
- No worst-case guarantee

- **Augmenting Learned + Random**
  ✓ Worst-case theoretical bounds
  ✓ Better empirical performance
Empirical Evaluation

Datasets:
1. Videos: MIT Logo, Friends, Eagle  
2. Hyperspectral images (HS-SOD)  
3. TechTC-300

- **Training phase**: 200/400 matrices.  
- **Testing phase**: 100 matrices

- Compare empirical recovery error $\sum_i ||A_i - SCW(S, A_i)||_F - ||A_i - [A_i]_k||_F$ of our learned matrix $S$ to random matrix $R$