1. **Numerical Linear Algebra in the Streaming Model**

Ken Clarkson  
IBM Almaden  

*joint with David Woodruff*

2. **The Input Data**

- $A$ is an $n \times d$ matrix, $B$ is $n \times d'$
- Matrix entries are given as a sequence of updates
- An update specifies $i$, $j$, $v$, and $A$ or $B$, so that $A_{ij} \leftarrow A_{ij} + v$, or similarly for $B$
  - The *turnstile* streaming model
- This is even more demanding than taking one pass over $A$ and $B$ fixed in memory

3. **The General Algorithmic Approach**

- As updates appear: maintain compressed versions of $A$ and $B$
  - *Sketches*
- When ready: compute output results using sketches
- Key resources: passes (=1 here), space, update time, compute time

4. **The Problems**

We give provably good estimators for:

- **Product:** $A^TB$
- **Regression:** the matrix $X^*$ minimizing $\|AX - B\|$
  - A slightly generalized version of least-squares regression
  - All norms here Frobenius, so $\|A\| := [\sum_{i,j}A_{ij}^2]^{1/2}$
- **Low Rank Approximation:** the matrix $A_k$ of rank $k$ minimizing $\|A - A_k\|$
  - For $k$ given beforehand
- The rank of $A$
5. General Properties of Our Algorithms

- Provable error bounds, with high probability
- The error is measured using the Frobenius norm
- For some problems, our sketches as small as possible
  - For a given error
  - When $A$ and $B$ have appropriate-sized integer entries
- Sketches may also be useful in a distributed setting, where matrix entries are scattered
  - ...and one pass $\Rightarrow$ few rounds of communication

6. Randomized Matrix Compression

In a line of similar efforts...
- Elementwise sampling [AM01][AHK06]
- Row/column sampling: pick small random subsets of the rows, columns, or both [DK01][DKM04]
  - Sample probability based on Euclidean norm of row or column
  - Or even: probability based on norm of vector in SVD
  - In general, needs two passes
  - Whole row or column samples are good "examples", and may preserve sparsity
- (Here) Sketching/Random Projection: maintain a small number of random linear combinations of rows or columns [S06]
- Our upper bound work is $\approx$ a followup to [S06]
  - cf. Rokhlin-Szlam-Tygert, Halko-Martinsson-Tropp

7. Approximate Matrix Product

- $A$ and $B$ have $n$ rows, we want to estimate $A^TB$
- Let $S$ be an $n \times m$ sign matrix
  - A.K.A. Rademacher or Bernoulli
  - Each entry is $+1$ or $-1$ with probability $1/2$
  - $m = O(1)$, to be specified
  - Independent entries, for now
- Our estimate of $A^TB$ is $A^TS^T B / m = (S^T A)^T S^T B / m$
- That is, sketches are $S^T A$ and $S^T B$
  - Compressing the columns from $n$ down to $m$
8. **Time and Space Bounds**

- Update time is $O(m)$, since only one column of $S^T$ is needed per update
- Space is $O(md)$ for $S^T A$, $O(md')$ for $S^T B$
  - $O(m)$ space for $S$, via limiting independence of $S$ entries
- Compute time, for product of sketches, is $O(mdd') = O(mc^2)$, $c := d + d'$
  - Can be done in $O(dd')$ [Coppersmith]
  - That is, we have optimal space, number of passes, and compute time

9. **Expected Error, and a Tail Estimate**

- From $E[SS^T]/m = I$ and linearity of expectation,

  $$E[A^TSS^TB/m] = A^T E[SS^T] B/m = A^T B$$

- So in expectation, sketch product is a good estimate of the product
- This is true also with high probability
- That is, for $\delta, \epsilon > 0$, there is $m = O(\epsilon^{-2} \log(1/\delta))$ so that

  $$\text{Prob}\{|A| > \epsilon|A||B|\} \leq \delta$$

  - Here $\Lambda$ is the error $A^TSS^TB/m - A^T B$
- This tail estimate seems to be new
  - Bound holds when entries of $S$ are $O(\log(1/\delta))$-wise independent

10. **Lower Bound on Space**

- The sketch size $O(M\epsilon^{-2} \log(1/\delta))$ is only a log $c$ factor improvement, $c = d + d'$
  - Entries are $M = O(\log(nc))$ bit integers
- However: the new upper bound matches our new space lower bound $\Omega(Mc/\epsilon^2)$
  - Failure probability $\delta \leq 1/4$
  - Large enough $n$ and $c$
11. Framework of Proof of Lower Bound

- Reduction from a communication task
  - Alice has random $x \in \{0, 1\}^s$
  - Bob has random $i$
  - Alice must send data to Bob so that he can learn $x_i$
- For even $2/3$ chance of success, Alice must send $\Omega(s)$ bits
  - Even when Bob already knows $x_{i'}$ for $i' > i$ [MNSW]
- Given a product algorithm using small sketches:
  - Alice can encode $x$ in $A$, send sketch of $A$ to Bob
  - Bob can use $B$ and sketch of $A$ to estimate $A^TB$, and find $x_i$

12. Regression

- The problem again: $\min_X \|AX - B\|^2$
- $X^*$ minimizing this has $X^* = A^- B$, where $A^-$ is the pseudo-inverse of $A$
- The algorithm is:
  - Maintain $S^TA$ and $S^TB$
  - Return $\hat{X}$ solving $\min_X \|S^T(AX - B)\|$
- Main claim: if $A$ has rank $k$,
  there is $m = O(k\epsilon^{-1}\log(1/\delta))$ so that with probability at least $1 - \delta$
  $\|A\hat{X} - B\| \leq (1 + \epsilon)\|AX^* - B\|$
  - That is, relative error for $\hat{X}$ is small
13. Regression Analysis Ideas

- Why should $\hat{X}$ be so good?
  - For fixed $Y$, $||S^T(AY-B)|| \approx ||AY-B||$
    - Just as for a random projection
  - If the norm is preserved for all $Y$, we're done
  - $S^T$ must preserve norm even of $\hat{X}$, chosen using $S$
- The main idea: show that $||S^T A (X^* - \hat{X})||$ is small
  - Using normal equations of sketched problem, matrix mult. results
- Use this to show $||A(X^* - \hat{X})||$ is small
- Use this to show the result
  - Using normal equations of exact problem

14. Best Low-Rank Approximation

- For any matrix $A$ and integer $k$, there is a matrix $A_k$ of rank $k$ that is closest to $A$ among all matrices of rank $k$
- Since rank of $A_k$ is $k$, it is the product $CD^T$ of two $k$-column matrices $C$ and $D$
  - $(A_k$ can be found from the SVD (singular value decomposition), where $C$ and $D$ are orthogonal matrices $U$ and $V\Sigma$)
  - This is a good compression of $A$
  - If entries of $A$ are noisy measurements, often the noise is "compressed out" in this way
  - LSI, PCA, Eigen*, recommender systems, clustering,...

15. Best Low-Rank Approximation and $S^T A$

- The sketch $S^T A$ holds a lot of information about $A$
- In particular, there is a rank $k$ matrix $\hat{A}_k$ in the rowspace of $S^T A$ nearly as close to $A$ as $A_k$
  - The rowspace of $S^T A$ is the set of linear combinations of its rows
- That is, $||A - \hat{A}_k|| \leq (1 + \epsilon) ||A - A_k||$
- This is shown using the regression results
16. Nearly Best Nearly-Low-Rank Approximation

- A similar observation applies in transpose
- Suppose $R$ is a $d \times m$ sign matrix (recall $A$ is $n \times d$)
- The columnspace of $AR$ contains a nearly best rank-$k$ approximation to $A$
- That is, $\hat{X}$ minimizing $\|ARX - A\|$ has $\|AR\hat{X} - A\| \leq (1 + \epsilon)\|A - A_k\|$
- Now minimize sketched version $\|S^T ARX - S^T A\|$
- Solution is $X' = (S^T AR)^- S^T A$ with
  $$\|ARX' - A\| \leq (1 + \epsilon)\|AR\hat{X} - A\| \leq (1 + \epsilon)^2\|A - A_k\|$$
  - Since $AR$ has rank $k\epsilon^{-1}$, $S$ must be $n \times m'$, with $m' = k\epsilon^{-2}$

17. Nearly Best Nearly-Low-Rank Algorithm

- An algorithm: maintain $AR$ and $S^T A$, return $ARX' = AR(S^T AR)^- S^T A$
  - Rank is $k/\epsilon$
  - Distance to $A$ is $(1 + \epsilon)\|A - A_k\|$
- This approximation to $A$ is interesting in its own right
  - No SVD required, only pseudo-inverse of a matrix of constant size

18. Nearly Best Low-Rank Approximation

Still haven't found a good rank $k$ matrix

- To do this, we find the best rank-$k$ approximation to $\frac{AR(S^T AR)^- S^T A}{A}$ in the columnspace of $AR$
- The resulting upper bound on space is a bigger w.r.t. than our lower bound
- When $A$ is given a column at a time, or a row at a time, we can do better
19. **Concluding Remarks**

- Space bounds are tight for product, regression
  - Faster update times?
- Space bounds are not tight w.r.t. $\epsilon$ for low-rank approximation
  - Upper bounds are at fault, probably
  - We have better upper bounds for restricted cases
- The entry-wise $r$-norm of the error matrix $\Lambda$ can also be bounded
  - This implies a bound on $||\Lambda||_{\max}$ in terms of $||A||_{1\to 2}$ and $||B||_{1\to 2}$
- Other projection matrices besides sign matrices?
- For what other problems is the full power of the JL transform not needed?

Thank you for your attention