Low Rank Approximation and Regression in Input Sparsity Time

David Woodruff
IBM Almaden

Joint work with Ken Clarkson (IBM Almaden)
Talk Outline

• Least-Squares Regression
  – Known Results
  – Our Results

• Low-Rank Approximation
  – Known Results
  – Our Results

• Experiments
Least-Squares Regression

• A is an n x d matrix, b an n x 1 column vector

• Consider over-constrained case, n ≥ d

• Find x so that $|Ax-b|_2 \cdot (1+\varepsilon) \min_y |Ay-b|_2$

• Allow a tiny probability of failure (depends only on randomness of algorithm, not on the input)
The Need for Approximation

• For $y = A \cdot b$, $Ay$ is the “closest” point in the column space of $A$ to the vector $b$

• Computing $y$ exactly takes $O(nd^2)$ time

• Too slow, so we allow $\epsilon > 0$ and a tiny probability of failure
Subspace Embeddings

• Let $k = O(d/\varepsilon^2)$
• Let $S$ be a $k \times n$ matrix of i.i.d. normal $N(0,1/k)$ random variables
• For any fixed $d$-dimensional subspace, i.e., the column space of an $n \times d$ matrix $A$ – W.h.p., for all $x$ in $\mathbb{R}^d$, $|SAx|_2 = (1 \pm \varepsilon)|Ax|_2$
• Entire column space of $A$ is preserved

Why is this true?
Subspace Embeddings – A Proof

• Want to show $|S Ax|_2 = (1 \pm \varepsilon)|Ax|_2$ for all $x$
• Can assume columns of $A$ are orthonormal (since we prove this for all $x$)
• By rotational invariance, $SA$ is a $k \times d$ matrix of i.i.d. $N(0, 1/k)$ random variables
• Well-known that singular values of $SA$ are all in the range $[1-\varepsilon, 1+\varepsilon]$
• Hence, $|S Ax|_2 = (1 \pm \varepsilon)|Ax|_2$

What does this have to do with regression?
Subspace Embeddings for Regression

- Want $x$ so that $|Ax-b|_2 \cdot (1+\varepsilon) \min_y |Ay-b|_2$
- Consider subspace $L$ spanned by columns of $A$ together with $b$
- Then for all $y$ in $L$, $|Sy|_2 = (1\pm \varepsilon) |y|_2$
- Hence, $|S(Ax-b)|_2 = (1\pm \varepsilon) |Ax-b|_2$ for all $x$
- Solve $\arg\min_y |(SA)y - (Sb)|_2$
- Given $SA$, $Sb$, can solve in $\text{poly}(d/\varepsilon)$ time

But computing $SA$ takes $O(nd^2)$ time right?
Subspace Embeddings - Generalization

• S need not be a matrix of i.i.d normals
• Instead, a “Fast Johnson-Lindenstrauss matrix” S suffices
• Usually have the form: $S = P*H*D$
  – D is a diagonal matrix with +1, -1 on diagonals
  – H is the Hadamard transform
  – P just chooses a random (small) subset of rows of $H*D$
• SA can be computed in $O(nd \log n)$ time
Previous Work vs. Our Result

- [AM, DKM, DV, …, Sarlos, DMM, DMMW, KN]
  Solve least-squares regression in $O(nd \log d) + \text{poly}(d/\epsilon)$ time

- Our Result
  Solve least-squares regression in $O(\text{nnz}(A)) + \text{poly}(d/\epsilon)$ time,
  where $\text{nnz}(A)$ is number of non-zero entries of $A$

  Much faster for sparse $A$, e.g., $\text{nnz}(A) = O(n)$
Our Technique

• Better subspace embedding!
• Define $k \times n$ matrix $S$, for $k = \text{poly}(d/\epsilon)$
• $S$ is really sparse: single randomly chosen non-zero entry per column

\[
\begin{bmatrix}
0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 \\
1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & -1 & 1 & 0 & -1 & 0 & 0 \\
0 & -1 & 0 & 0 & 0 & 0 & 0 & 1
\end{bmatrix}
\]
Surprising Part

- For certain $k = \text{poly}(d/\epsilon)$, w.h.p., for all $x$,
  \[ |SAx|_2 = (1 \pm \epsilon) |Ax|_2 \]

- Since $S$ is so sparse, $SA$ can be computed in $\text{nnz}(A)$ time

- Regression can be solved in $\text{nnz}(A) + \text{poly}(d/\epsilon)$ time
Why Did People Miss This?

- Usually put a net on a d-dimensional subspace, and argue for all $z$ in the net,
  $$|SAz|_2 = (1 \pm \epsilon) |Az|_2$$

- Since the net has size $\exp(d)$, need $S$ to preserve the lengths of $\exp(d)$ vectors

- If these vectors were arbitrary, the above $S$ would not work!

So how could this possibly work?
Leverage Scores

• Suffices to prove for all unit vectors $x$
  \[ |S Ax|_2 = (1 \pm \varepsilon) |Ax|_2 \]

• Can assume columns of $A$ are orthonormal
  – $|A|_F^2 = d$

• Let $T$ be any set of size $d/\bar{\nu}$ containing all $i \in [n]$ for which $|A_i|^2 \geq \bar{\nu}$
  – $T$ contains the large leverage scores

• For any unit $x$ in $\mathbb{R}^d$,
  \[ |(Ax)_i| = |<A_i, x>| \cdot |A_i|_2 \leq |x|_2 \cdot |A_i|_2 \]

• Say a coordinate $i$ is heavy if $|(Ax)_i|^2 \geq (1 - \varepsilon) |Ax|_2^2$
  – Heavy coordinates are a subset of $T$!
Perfect Hashing

- View map $S$ as randomly hashing coordinates into $k$ buckets, and maintaining an inner product with a sign vector in each bucket.

$$
\begin{bmatrix}
0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 \\
1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & -1 & 1 & 0 & -1 & 0 \\
0 & -1 & 0 & 0 & 0 & 0 & 0 & 1 \\
\end{bmatrix}
$$

- If $k > 10d^2/\sqrt{2} = 10 |T|^2$, then with constant probability, all coordinates in $T$ are perfectly hashed.
- Call this event $E$ and condition on $E$.
The Three Error Terms

• Suppose $y = Ax$ for an $x$ in $\mathbb{R}^d$

• $y = y_T + y_{[n]\setminus T}$

• $|Sy|_2^2 = |Sy_T|_2^2 + |Sy_{[n]\setminus T}|_2^2 + 2<Sy_T, Sy_{[n]\setminus T}>$
The Large Coordinate Error Term

• Need to bound $|S_y_T|^2$

• Since event $E$ occurs, $|S_y_T|^2 = |y_T|^2$
The Small Coordinate Error Term

• Need to bound $|Sy_{[n]\mathbb{T}}|^2$

• Key point: $|y_{[n]\mathbb{T}}|_1$ is small

• [DKS]: There is an $\ominus \frac{1}{4} \varepsilon^2/d$ so that if $k = \Omega(\log(1/\delta)/\varepsilon^2)$ for a mapping of our form $S$, then for any vector $y$ with $|y|_1 = O(\ominus)$,
  \[ \Pr[|Sy|^2 = |y_{[n]\mathbb{T}}|^2 \pm O(\varepsilon)] = 1 - O(\delta) \]

• Set $\tilde{=} = O(\ominus) = 1/poly(d/\varepsilon)$ so $|y_{[n]\mathbb{T}}|_1 = O(\tilde{\ominus})$

• Hence, $\Pr[|Sy_{[n]\mathbb{T}}|^2 = |y_{[n]\mathbb{T}}|^2 \pm O(\varepsilon)] = 1 - O(\delta)$
The Cross-Coordinate Error Term

- Need to bound $|<S_y_T, S_y_{[n]\bar{T}}>|$
- $S_y_T$ only has support on $|T|$ coordinates
- Let $G \mu [n]\bar{T}$ be such that each $i \in G$ hashes to a bucket containing a $j \in 2 T$
- $|<S_y_T, S_y_{[n]\bar{T}}>| = |<S_y_T, S_y_G>| \cdot |S_y_T|_2 \cdot |S_y_G|_2$
- $|S_y_T|_2 = |y_T|_2 \cdot 1$ by event $E$
- $\Pr[|S_y_G|_2 \cdot |y_G|_2 + O(\varepsilon)] = 1-O(\delta)$ by [DKS]
- $\Pr[|y_G|_2 \cdot \varepsilon] = 1-O(\delta)$ by Hoeffding
- Hence, $\Pr[|<S_y_T, S_y_{[n]\bar{T}}>| \cdot 2\varepsilon] = 1-O(\delta)$
Putting it All Together

- Given that event E occurs, for any fixed y, with probability at least 1-\(O(\delta)\):

\[
|Sy|_2^2 = |Sy_T|_2^2 + |Sy_{[n]\setminus T}|_2^2 + 2\langle Sy_T, Sy_{[n]\setminus T}\rangle \\
= |y_T|_2^2 + |y_{[n]\setminus T}|_2^2 \pm O(\varepsilon) \\
= |y|_2^2 \pm O(\varepsilon) \\
= (1 \pm O(\varepsilon))|y|_2^2
\]
The Net Argument

[F, M, AHK]: If for any fixed pair of unit vectors $x,y$, a random $d \times d$ matrix $M$ satisfies

$$\Pr[|x^T M y| = O(\varepsilon)] > 1-\exp(-d),$$

then for every unit vector $x$, $|x^T M x| = O(\varepsilon)$

- We apply this to $M = (SA)^T SA - I_d$
- Set $\delta = \exp(-d)$:
  - For any $x,y$ with probability $1-\exp(-d)$:
    $$|SA(x+y)|_2 = (1\pm\varepsilon)|A(x+y)|_2$$
    $$|SAx|_2 = (1\pm\varepsilon)|Ax|_2,$$
    $$|SAy|_2 = (1\pm\varepsilon)|Ay|_2$$

Hence, $|x^T M y| = O(\varepsilon)$
Talk Outline

• Least-Squares Regression
  – Known Results
  – Our Results

• Low-Rank Approximation
  – Known Results
  – Our Results

• Experiments
Low Rank Approximation

A is an n x n matrix

Want to output a rank k matrix $A'$, so that

$$|A-A'|_F \cdot (1+\varepsilon) |A-A_k|_F,$$

w.h.p., where $A_k = \text{argmin}_{\text{rank } k \text{ matrices } B} |A-B|_F$

Previous results:

$$\text{nnz}(A)*(k/\varepsilon + k \log k) + n*\text{poly}(k/\varepsilon)$$

Our result: $\text{nnz}(A) + n*\text{poly}(k/\varepsilon)$
Technique

- [CW] Let $S$ be an $n \times k/\epsilon^2$ matrix of i.i.d. $\pm 1$ entries, and $R$ an $n \times k/\epsilon$ matrix of i.i.d. $\pm 1$ entries. Let $A' = AR(S^TAR)^{-1} S^T A$.
- Can extract low rank approximation from $A'$

- **Our result:** similar analysis works if $R, S$ are our new subspace embedding matrices
- Operations take $\text{nnz}(A) + n \times \text{poly}(k/\epsilon)$ time
Talk Outline

• Least-Squares Regression
  – Known Results
  – Our Results

• Low-Rank Approximation
  – Known Results
  – Our Results

• Experiments
Experiments

• Looked at low rank approximation
  – $n \leq 600$, $\text{nnz}(A)$ at most $10^5$
• Test matrices from University of Florida Sparse Matrix Collection
• 40 different sparsity patterns, representing different application areas
• 500 different matrices
• Dominant time is computing $SA$, takes same time as one matrix-vector product in Lanczos
Experiments

\[ R_e = \text{sketch err / rank-err} \]

Graph showing the relationship between \( t/k \) and \( R_e \) with steps at specific values of \( t/k \).
Conclusions

• Gave new subspace embedding of a d-dimensional subspace of $\mathbb{R}^n$ in time:
  $$\text{nnz}(A) + \text{poly}(d/\epsilon)$$

• Achieved the same time for regression, improving prior $nd \log d$ time algorithms

• Achieved $\text{nnz}(A) + n\times\text{poly}(k/\epsilon)$ time for low-rank approximation, improving previous $nd \log d + n\times\text{poly}(k/\epsilon)$ time algorithms