CS 15-859: Algorithms for Big Data Fall 2021

Lecture 4.2 — 09/30

Prof. David Woodruff *Scribe: Chittesh Thavamani*

1 Fast Approximation of Leverage Scores

We previously showed that leverage score sampling yields a valid subspace embedding. Now, given an $n \times d$ matrix A, we just need a fast method to approximate its leverage scores l_i for $i \in [n]$.

Naively, we could compute the SVD $A = U\Sigma V^\top$ and since the columns of U form an orthonormal basis for the column space of A, the leverage scores of A are the row norms of U. However, this would take poly(n) time, which is too slow.

Instead, we perform the following process:

(1) Compute SA for a subspace embedding S using a CountSketch matrix with $O(d^2/\varepsilon^2)$ rows for any chosen constant ε.

(2) Compute the thin SVD $SA = U_t\Sigma_t V_t^\top$. This yields a QR decomposition $SA = QR^{-1}$, where $Q = U_t$ has orthonormal columns and $R^{-1} = \Sigma_t V_t^\top$. Compute $R = V_t\Sigma_t^{-1}$.

(3) Sample a $d \times O(\log n)$ matrix G of i.i.d. normals. Compute ARG by first computing RG and then left multiplying by A.

(4) Output the squared row norms of ARG.

1.1 Correctness

Let $l_i' = |e_i AR|_2^2$ and $l_i'' = |e_i ARG|_2^2$. In other words, l_i' and l_i'' are the squared row norms of AR and ARG, respectively. We'll show that both these quantities are constant factor approximations of l_i.

Claim 1. For all $i \in [n]$, $l_i = (1 \pm O(\varepsilon))l_i'$.

Proof. Note that since R is invertible, the column space of AR is the same as A. And A has the same column space as U, where $A = U\Sigma V^\top$ is its SVD. So we must have $AR = UT^{-1}$ for some change of basis matrix T^{-1}.

Now, since AR has the same column space as A, we can apply the subspace embedding property on AR. Thus, for all vectors $x \in \mathbb{R}^d$

$$\begin{aligned}
(1-\varepsilon)|ARx|_2 &\le |SARx|_2 && \text{(subspace embedding property)}\\
&= |Qx|_2 && \text{(def of QR decomposition)}\\
&= |x|_2 && (Q \text{ has orthonormal cols})
\end{aligned}$$

Similarly, $(1+\varepsilon)|ARx|_2 \geq |x|_2$. Combining these, we have that

$$\begin{aligned}(1 \pm O(\varepsilon))|x|_2 &= |ARx|_2 \\ &= |UT^{-1}x|_2 && \text{(def of } T^{-1}\text{)} \\ &= |T^{-1}x|_2 && (U \text{ is orthonormal})\end{aligned}$$

This means that T^{-1} (and its inverse T) preserves the lengths of vectors $x \in \mathbb{R}^d$ up to a factor of $1 \pm O(\varepsilon)$.

Finally, we have that

$$\begin{aligned}l_i &= |e_i U|_2^2 && \text{(def of leverage score)} \\ &= |e_i ART|_2^2 && \text{(def of } T\text{)} \\ &= (1 \pm O(\varepsilon))|e_i AR|_2^2 && (T \text{ preserves lengths approximately}) \\ &= (1 \pm O(\varepsilon))l_i'.\end{aligned}$$

Thus, since ε is a chosen constant, l_i' is a constant factor approximation of l_i. ■

Theorem 1. *Johnson-Lindenstrauss: Let G be a $d \times O(\log n)$ matrix of i.i.d. normals. Then for any vector z,*

$$\mathbf{Pr}\Big[|zG|_2^2 = \left(1 \pm \frac{1}{2}\right)|z|_2^2\Big] \geq 1 - 1/n^2.$$

Claim 2. For all $i \in [n]$, $l_i'' = (1 \pm 1/2)l_i'$.

By the previous theorem, we have for any $i \in [n]$ that $l_i'' = |e_i ARG|_2^2 = (1 \pm 1/2)|e_i AR|_2^2 = (1 \pm 1/2)l_i'$ with failure probability $1/n^2$. By union bound, this is satisfied for all $i \in [n]$ with failure probability at most $1/n$. ■

Thus, putting our claims together, our output l_i'' is a constant factor approximation of l_i, which is exactly what we want.

1.2 Runtime

Step 1 takes $\text{nnz}(A)$ time since S is a CountSketch matrix.

SA is $O(d^2/\varepsilon^2) \times d$, so computing its thin SVD takes $\text{poly}(d)$ time. From there, computing R will also take $\text{poly}(d)$ time.

R is $d \times d$ and G is $d \times O(\log n)$, so RG takes $O(d^2 \log n)$ time to compute. Then, left multiplying by A involves multiplying each nonzero element of A with the corresponding row of RG with $O(\log n)$ elements, so this take $\text{nnz}(A) \log n$ time.

Outputting the squared row norms of ARG of size $n \times O(\log n)$ takes $O(\text{nnz}(A) \log n)$ time, since $\text{nnz}(A) \geq n$.

Finally, sampling according to leverage scores results and solving that least squares problem will take $\text{poly}(d \log n/\varepsilon)$ time.

So the overall runtime is $O(\text{nnz}(A) \log n + \text{poly}(d \log n/\varepsilon))$.

2 Distributed Low Rank Approximation

Can we solve the low rank approximation problem efficiently in a distributed setting (the data is distributed over multiple machines) where communication between machines is an additional constraint? One real world application of this could be k-means clustering, in which we might have too many data points to fit onto one machine, and a common approach is to project these points onto a lower dimensional space before performing the clustering.

2.1 Problem formulation

Let A be the matrix whose low rank k approximation we want to find. We have s servers, each holding a part A^i of the matrix A for $i \in [s]$. Either we have the *arbitrary partition model*, in which case $A = A^1 + A^2 + \ldots + A^s$, or we have the *row partition model*, in which case $A = \left[\begin{array}{c|c|c|c} A^1 & A^2 & \cdots & A^s \end{array} \right]^\top$. Note that the arbitrary partition model is a generalization of the row partition model. Assume all entries of A^i for $i \in [s]$ are $O(\log(nd))$-bit integers (WLOG we can scale up floating point values and round to the nearest integer).

Each server can communicate both ways with a specified other "coordinator" machine. This still allows us to simulate arbitrary server to server communication up to a factor of 2 in cost (and an additive factor of $O(\log s)$ to specify target machine) just by relaying all messages through the coordinator node.

The goal is for each server to output the same k-dimensional space W such that

$$|A - C|_F \leq (1 + \varepsilon)|A - A_k|_F$$

where $C = A^1 P_W + A^2 P_W + \ldots A^s P_W = A P_W$ is the combined low rank approximation for P_W a projection onto W and A_k is the optimal low rank approximation. Our objectives are to minimize total communication and computation. We also want a constant number of rounds and input sparsity time.

2.2 Previous Results

1. FSS [2] provides a solution to the row-partition model using $O(sdk/\varepsilon)$ real numbers of communication. Disadvantages of this method is that the bit complexity of communication is unaddressed. The assumption of being able to send real numbers might not hold in real life. Also, each server requires an SVD to be performed on its data, which is expensive.

2. KVW [3] provides a solution to the arbitrary partition model using $O(sdk/\varepsilon)$ communication.

3. BWZ [1] provides a solution to the arbitrary partition model using $O(sdk) + \text{poly}(sk/\varepsilon)$ communication and input sparsity time for computation. Note that this matches the $\Omega(sdk)$ lower bound that's intuitively required for sharing a $d \times k$ matrix across s servers. This can be proven formally.

2.3 FSS: Constructing a Coreset

The first step to the FSS algorithm is constructing what is known as a coreset of A. Let $A = U\Sigma V^\top$ be its SVD, and define $m = k + k/\varepsilon$. Define Σ_m to be the same as Σ except with zeros past the first m diagonal entries. Similarly, define U_m to be the same as U except with zeros beyond the first m columns. Finally, $A_m = U\Sigma_m V^\top$.

Claim 3. For all projection matrices $Y = I - X$ onto $(d-k)$-dimensional subspaces,

$$|\Sigma_m V^\top Y|_F^2 + c = (1 \pm \varepsilon)|AY|_F^2$$

where $c = |A - A_m|_F^2$ notably doesn't depend on Y. We call $\Sigma_m V^\top$ the coreset of A.

Intuition: Multiplying a vector v by F results in $vF = v - vX$. Since X is another projection matrix, $|vF|_2$ corresponds to the distance from v to the subspace represented by X. Thus, $|AY|_F^2$ is equal to the sum of squared distances from the row vectors of A to the subspace represented by X. Similarly, $|\Sigma_m V^\top Y|_F^2$ is the sum of squared distances from the row vectors of the coreset to the subspace represented by X. The claim is stating that these two quantities are roughly equal for any choice of projection Y, up to the constant c that is invariant to Y.

We can also think of U_m as a sketching matrix S here, because $SA = U_m U\Sigma V^\top = \Sigma_m V^\top$.

Proof. First, we'll show that $|\Sigma_m V^\top Y|_F^2 + c \geq (1-\varepsilon)|AY|_F^2$ by showing the stronger result that $|AY|_F^2 \leq |\Sigma_m V^\top Y|_F^2 + c$.

For a projection matrix P and a matrix M, we have that $|M|_F^2 = |PM|_F^2 + |(I-P)M|_F^2$ by use of the Pythagorean Theorem on the columns of M. Now, consider $P = U_m U_m^\top$ and $M = AY$. This gives us

$$\begin{aligned} |AY|_F^2 &= |U_m U_m^\top AY|_F^2 + |(I - U_m U_m^\top)AY|_F^2 \\ &= |U_m U_m^\top AY|_F^2 + |(A - U_m U_m^\top A)Y|_F^2 \end{aligned}$$

Here, by the SVD of A, we have that $U_m U_m^\top A = U_m U_m^\top U\Sigma V^\top = U_m I_m \Sigma V^\top = U_m \Sigma_m V^\top = U\Sigma_m V^\top$, where I_m is an identity matrix with zeros as diagonal entries past the mth. It follows that

$$\begin{aligned} |AY|_F^2 &= |U\Sigma_m V^\top Y|_F^2 + |(A - U\Sigma_m V^\top)Y|_F^2 \\ &= |\Sigma_m V^\top Y|_F^2 + |(A - A_m)Y|_F^2 && (U \text{ is orthonormal, def of } A_m) \\ &\leq |\Sigma_m V^\top Y|_F^2 + |A - A_m|_F^2 \\ &\qquad \text{(projecting through } Y \text{ would only decrease row norms of } A - A_m) \\ &= |\Sigma_m V^\top Y|_F^2 + c. \end{aligned}$$

Now, we'll show that $|\Sigma_m V^\top Y|_F^2 + c \leq (1+\varepsilon)|AY|_F^2$ by showing that $|\Sigma_m V^\top Y|_F^2 + c - |AY|_F^2 \leq \varepsilon |AY|_F^2$.

By the Pythagorean Theorem, by the definition of projection matrix $Y = I - X$, for any vector v, we have that $|v|_2^2 = |vY|_2^2 + |vX|_2^2 \implies |vY|_2^2 = |v|_2^2 - |vX|_2^2$. Extending this to

matrices M of stacked row vectors, we have that $|MY|_F^2 = |M|_F^2 - |MX|_2^2$. It follows that

$$\begin{aligned}
&|\Sigma_m V^\top Y|_F^2 + |A - A_m|_F^2 - |AY|_F^2 \\
=&|\Sigma_m V^\top|_F^2 - |\Sigma_m V^T X|_F^2 + |A - A_m|_F^2 - |A|_F^2 + |AX|_F^2 && \text{(previous result)} \\
=&|U\Sigma_m V^\top|_F^2 - |\Sigma_m V^T X|_F^2 + |A - A_m|_F^2 - |A|_F^2 + |AX|_F^2 && (U \text{ is orthonormal}) \\
=&|A_m|_F^2 - |\Sigma_m V^T X|_F^2 + |A - A_m|_F^2 - |A|_F^2 + |AX|_F^2. && \text{(def of } A_m)
\end{aligned}$$

Again, by the Pythagorean Theorem, letting $M = A$ and $P = U_m U_m^\top$ (notation as in proof of other direction), after some algebra, we have that $|A|_F^2 = |A_m|_F^2 + |A - A_m|_F^2$. It follows that

$$\begin{aligned}
&|\Sigma_m V^\top Y|_F^2 + |A - A_m|_F^2 - |AY|_F^2 \\
=&|AX|_F^2 - |\Sigma_m V^T X|_F^2 && \text{(prev Pythag thm result)} \\
=&|AX|_F^2 - |U\Sigma_m V^T X|_F^2 && (U \text{ is orthonormal}) \\
=&|AX|_F^2 - |A_m X|_F^2 && \text{(def of } A_m) \\
=&|(A - A_m)X|_F^2 && \text{(Pythag thm with } M = AX, P = U_m U_m^\top) \\
=&|U(\Sigma - \Sigma_m)V^\top X|_F^2 && \text{(factor SVD of } A \text{ and } A_m) \\
=&|(\Sigma - \Sigma_m)V^\top X|_F^2 && (U \text{ is orthonormal})
\end{aligned}$$

Here, we use a property called submultiplicativity which states that $|AB|_F^2 \leq |A|_2^2 \cdot |B|_F^2$, which can be proven by breaking up the Frobenius norm by columns of B and noting that the norm of each column will be increased by at more the operator norm of A. It follows that

$$|\Sigma_m V^\top Y|_F^2 + |A - A_m|_F^2 - |AY|_F^2 \leq |(\Sigma - \Sigma_m)V^\top|_2^2 \cdot |X|_F^2$$

Now, we have that the operator norm of $(\Sigma - \Sigma_m)V^\top$ is its max singular value, which is σ_{m+1}^2 or the $(m+1)$-th singular value of A. And since X is a projection matrix of rank k, its Frobenius norm must be k. Thus, it follows that

$$\begin{aligned}
&|\Sigma_m V^\top Y|_F^2 + |A - A_m|_F^2 - |AY|_F^2 \\
\leq&\sigma_{m+1}^2 k \\
=&\sigma_{m+1}^2 \varepsilon(m - k) && \text{(def of m)} \\
\leq&\varepsilon \sum_{i=k+1}^{m+1} \sigma_i^2 && \text{(include } m - k \text{ previous singular values)} \\
\leq&\varepsilon \sum_{i=k+1}^{d} \sigma_i^2 && \text{(include all singular values after } \sigma_{m+1}) \\
=&\varepsilon |A - A_k|_F^2 && \text{(squared Frobenius norm is the sum of squared singular values)} \\
\leq&\varepsilon |AY|_F^2. && (A - A_k \text{ is the projection of } A \text{ with minimum Frobenius norm})
\end{aligned}$$

Thus, our claim is proven.

This formulation is useful even with the additive constant c, because of the following argument.

Suppose $\tilde{Y}$ minimizes $|\Sigma_m V^\top Y|_F^2$ and Y^* minimizes $|AY|_F^2$. Then, we have that

$$\begin{aligned}
|A\tilde{Y}|_F^2 &\leq |\Sigma_m V^\top \tilde{Y}|_F^2 + c && \text{(stronger result proven in claim)} \\
&\leq |\Sigma_m V^\top Y^*|_F^2 + c && \text{(def of } \tilde{Y}) \\
&\leq (1+\varepsilon)|AY^*|_F^2 && \text{(claim)} \\
&= (1+\varepsilon)|A - A_k|_F^2. && \text{(the low rank solution } A_k \text{ is the projection that minimizes } |AY|_F^2)
\end{aligned}$$

This shows that we can reduce the low rank approximation problem to simply finding $\tilde{Y}$ using the "sketched" matrix.

References

[1] Christos Boutsidis, David P. Woodruff, and Peilin Zhong. *Optimal Principal Component Analysis in Distributed and Streaming Models*. 2016. arXiv: `1504.06729 [cs.DS]`.

[2] Dan Feldman, Melanie Schmidt, and Christian Sohler. *Turning Big data into tiny data: Constant-size coresets for k-means, PCA and projective clustering*. 2018. arXiv: `1807.04518 [cs.DS]`.

[3] Ravindran Kannan, Santosh Vempala, and David Woodruff. *Principal Component Analysis and Higher Correlations for Distributed Data*. 2014. arXiv: `1304.3162 [cs.DS]`.