COMPSCI 614: Randomized Algorithms with Applications to Data Science

Prof. Cameron Musco

University of Massachusetts Amherst. Spring 2024.

Lecture 11

Logistics

- The midterm is the Thursday after break.
- Tuesday that week I will do midterm review (not cover any new material).
- Midterm study material will be posted shortly.

Summary

Last Time:

- Approximate matrix multiplication via importance sampling.
- Application to fast low-rank approximation via sampling.

Today:

- Finish up fast low-rank approximation.
- Stochastic trace estimation.

Quiz Review

Question 6

Not complete

Points out of 1.00

Flag question

Edit question

Considering estimating the sum of a set of numbers $\{x_1, \ldots, x_n\}$ via importance sampling. You take t independent samples $i_1, \ldots, i_t$ according to probability distribution $\{p_1, \ldots, p_n\}$, multiply each sample by $1/p_{i_j}$, and then average together these samples.

Which of the following distributions minimizes the variance of your estimate. **To Think About:** Is sampling according to this probability distribution likely feasible?

a. $p_i = \frac{x_i}{\sum_{j=1}^{n} x_j}$

b. $p_i = \frac{\sqrt{x_i}}{\sum_{j=1}^{n} \sqrt{x_j}}$

c. $p_i = \frac{x_i^2}{\sum_{j=1}^{n} x_j^2}$

d. $p_i = 1/n$

Check

Quiz Review

Question 5

Not complete

Points out of 1.00

Flag question

Edit question

Consider a version of Count-Sketch where instead of returning the median of our t independent estimates as the final estimate, we return the estimate which is at the 95th percentile of the t estimates. How would this change the theoretical bounds for the algorithm? How do you think it would effect the practical performance?

a. We are not be able to obtain error $\epsilon\|x\|_2$ using this approach.

b. We could achieve the same error bounds and sketching dimension up to constants using this approach as compared to using the median. However, the approach will likely work significantly worse in practice.

c. We could achieve the same error bounds and sketching dimension up to constants using this approach as compared to using the median. Further, the approach will likely work significantly better in practice.

d. We could obtain the same error bounds using this approach, but would require a much larger sketching dimension as compared to taking the median. Further, the approach will likely work significantly worse in practice.

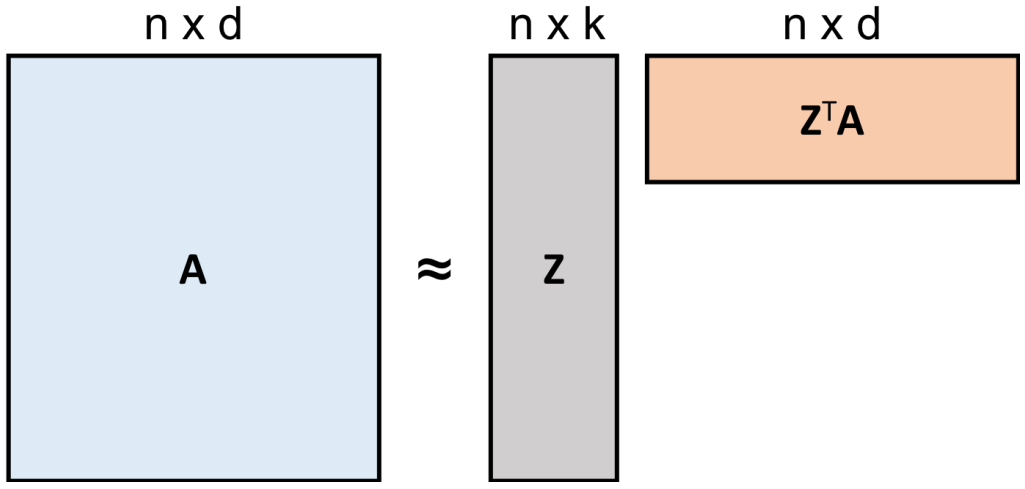
Check

Randomized Low-Rank approximation

Low-rank Approximation

Consider a matrix $A \in \mathbb{R}^{n \times d}$. We would like to compute an optimal low-rank approximation of A. I.e., for $k \ll \min(n, d)$ we would like to find $Z \in \mathbb{R}^{n \times k}$ with orthonormal columns satisfying:

$$\|A - ZZ^T A\|_F = \min_{Z : Z^T Z = I} \|A - ZZ^T A\|_F.$$



Solving this exactly requires computing the top k left singular vectors of A in $O(nd^2)$ time. We will give an approximation algorithm running in $O(nd + nk^2)$ time.

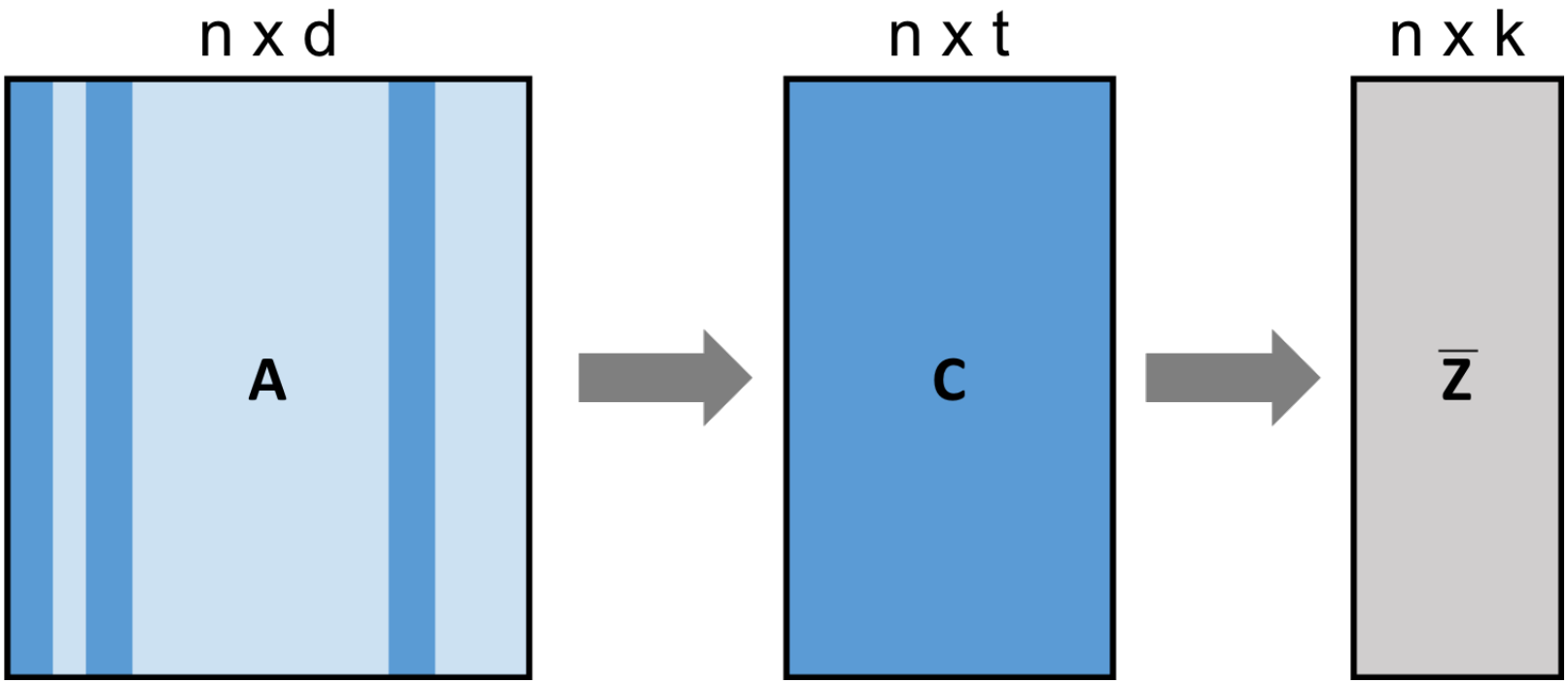
Sampling Based Algorithm

Linear Time Low-Rank Approximation:

- Fix sampling probabilities $p_1, \ldots, p_n$ with $p_i = \frac{\|A_{:,i}\|_2^2}{\|A\|_F^2}$.
- Select $\mathbf{i_1}, \ldots, \mathbf{i_t} \in [n]$ independently, according to the distribution $\mathbf{Pr}[\mathbf{i_j} = k] = p_k$ for sample size $t \geq k$.
- Let $\mathbf{C} = \frac{1}{t} \cdot \sum_{j=1}^{t} \frac{1}{\sqrt{p_{\mathbf{i_j}}}} \cdot A_{:,\mathbf{i_j}}$.
- Let $\overline{\mathbf{Z}} \in \mathbb{R}^{n \times k}$ consist of the top k left singular vectors of $\mathbf{C}$.

- By our approximate matrix multiplication analysis, if $t = O(\frac{k}{\delta \epsilon^2})$, $\|\mathbf{C}\mathbf{C}^T - AA^T\|_F \leq \epsilon/\sqrt{k} \cdot \|A\|_F^2$ with probability at least $1 - \delta$.
- We will use this to show that an optimal basis for $\mathbf{C}$ (i.e., $\overline{\mathbf{Z}}$) is nearly optimal for A.

Sampling Based Algorithm



Formal Analysis

Let $Z_* \in \mathbb{R}^{n \times k}$ contain the top left singular vectors of A – i.e. $Z_* = \textbf{arg min} \, \|A - ZZ^T A\|_F^2$. Similarly, $\overline{\mathbf{Z}} = \textbf{arg min} \, \|\mathbf{C} - ZZ^T \mathbf{C}\|_F^2$.

Claim 1: For any orthonormal $Z \in \mathbb{R}^{n \times k}$, and any matrix B,

$$\|B - ZZ^T B\|_F^2 = \textbf{tr}(BB^T) - \textbf{tr}(Z^T BB^T Z).$$

Claim 2: If $\|AA^T - \mathbf{C}\mathbf{C}^T\|_F \leq \frac{\epsilon}{\sqrt{k}} \|A\|_F^2$, then for any orthonormal $Z \in \mathbb{R}^{n \times k}$, $\textbf{tr}(Z^T(AA^T - CC^T)Z) \leq \epsilon \|A\|_F^2$.

Proof from claims:

$$\begin{aligned}
\|\mathbf{C} - \overline{\mathbf{Z}}\overline{\mathbf{Z}}^T \mathbf{C}\|_F^2 \leq \|\mathbf{C} - Z_* Z_*^T \mathbf{C}\|_F^2 \implies & \textbf{tr}(\overline{\mathbf{Z}}^T \mathbf{C}\mathbf{C}^T \overline{\mathbf{Z}}) \geq \textbf{tr}(Z_*^T \mathbf{C}\mathbf{C}^T Z_*) \\
\implies & \textbf{tr}(\overline{\mathbf{Z}}^T AA^T \overline{\mathbf{Z}}) \geq \textbf{tr}(Z_*^T AA^T Z_*) - 2\epsilon \|A\|_F^2 \\
\implies & \|A - \overline{\mathbf{Z}}\overline{\mathbf{Z}}^T A\|_F^2 \leq \|A - Z_* Z_*^T A\|_F^2 + 2\epsilon \|A\|_F^2.
\end{aligned}$$

Formal Analysis

Claim 2: If $\|AA^T - CC^T\|_F \leq \frac{\epsilon}{\sqrt{k}}\|A\|_F^2$, then for any orthonormal $Z \in \mathbb{R}^{n\times k}$, $\mathrm{tr}(Z^T(AA^T - CC^T)Z) \leq \epsilon\|A\|_F^2$.

Suffices to show that for any symmetric $B \in \mathbb{R}^{n\times n}$, and any orthonormal $Z \in \mathbb{R}^{n\times k}$, $\mathrm{tr}(Z^TBZ) \leq \sqrt{k} \cdot \|B\|_F$.

$$\begin{aligned}
\mathrm{tr}(Z^TBZ) &= \sum_{i=1}^{k} z_i^T B z_i \\
&\leq \sum_{i=1}^{k} \lambda_i(B) \qquad \text{(By Courant-Fischer theorem)} \\
&\leq \sqrt{k} \cdot \sqrt{\sum_{i=1}^{k} \lambda_i(B)^2} \leq \sqrt{k} \cdot \sqrt{\sum_{i=1}^{n} \lambda_i(B)^2} = \sqrt{k} \cdot \|B\|_F.
\end{aligned}$$

More Advanced Techniques

Norm based sampling gives an additive error approximation, $\|A - \overline{Z}\overline{Z}^T A\|_F^2 \leq \min_{Z:Z^TZ=I} \|A - ZZ^TA\|_F^2 + 2\epsilon\|A\|_F^2$.

- Ideally, we would like a relative error approximation, $\|A - \overline{Z}\overline{Z}^T A\|_F^2 \leq (1+\epsilon) \cdot \min_{Z:Z^TZ=I} \|A - ZZ^TA\|_F^2$.
- This can be achieved with more advanced non-uniform sampling techniques, based on leverage scores or adaptive sampling.
- Also possible using Johnson-Lindenstrauss type random projection.
- We will cover these techniques in future classes.

Stochastic Trace Estimation

Matrix Trace

The trace of a matrix $A \in \mathbb{R}^{n \times n}$ is the sum of it diagonal entries.

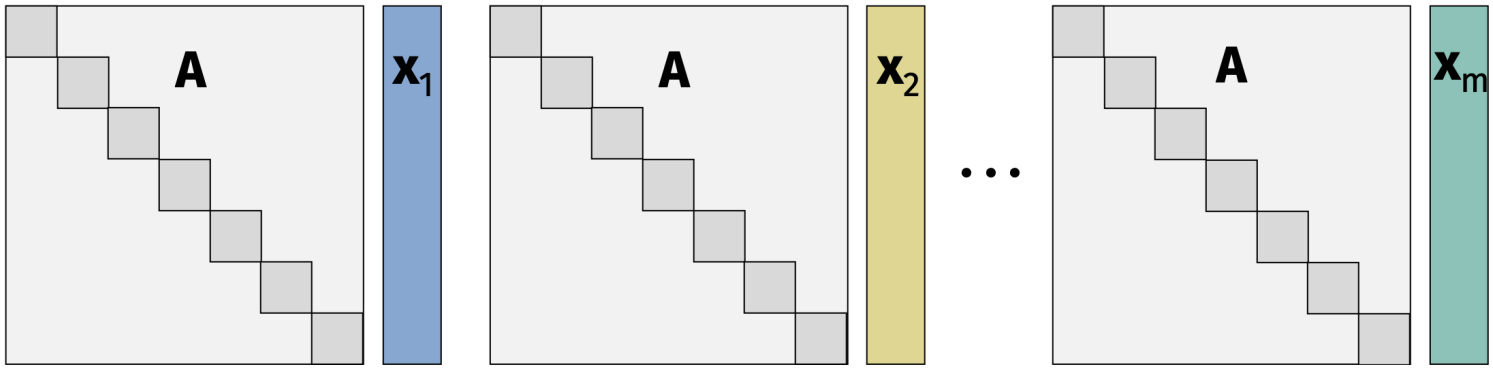
$$\mathbf{tr}(A) = \sum_{i=1}^{n} A_{ii}.$$

When A is diagonalizable (e.g., when it is symmetric) with eigenvalues $\lambda_1, \ldots, \lambda_n$, $\mathbf{tr}(A) = \sum_{i=1}^{n} \lambda_i$.

How many operations does it take to compute $\mathbf{tr}(A)$ given explicit access to A?

Implicit Trace Estimation

- Given implicit access to $A \in \mathbb{R}^{n \times n}$ through matrix-vector multiplication.
- Goal is to approximate $\mathbf{tr}(A) = \sum_{i=1}^{n} A_{ii}$.



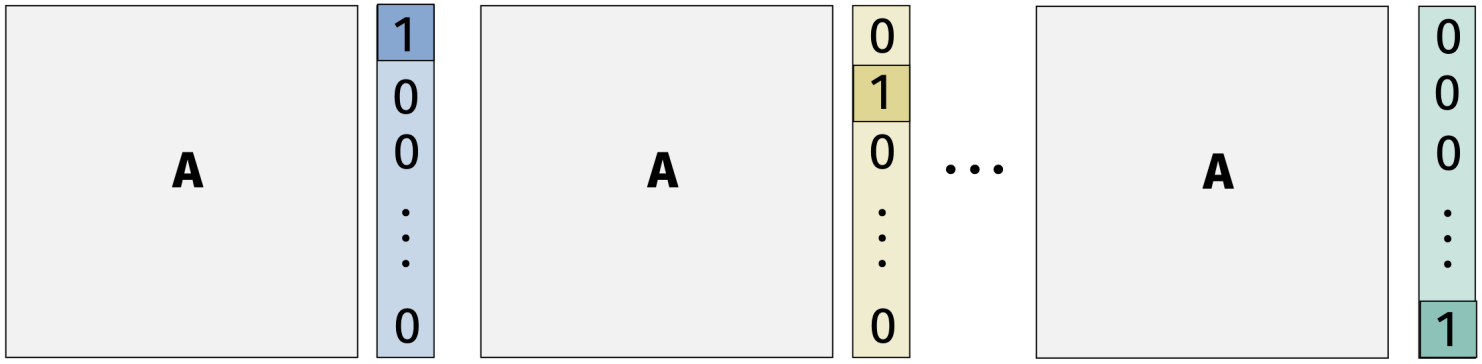
Main question: How many matrix-vector multiplication "queries" $Ax_1, \ldots, Ax_m$ are required to approximate $\mathbf{tr}(A)$?

Algorithms in this model are called matrix-free methods. Useful when A is not given explicitly, but we have an efficient algorithm for multiplying A by a vector (examples to come).

Naive Exact Algorithm

Naive solution:

- Set $x_i = e_i$ for $i = 1, \ldots, n$.
- Return $\mathbf{tr}(A) = \sum_{i=1}^{n} x_i^T A x_i$.

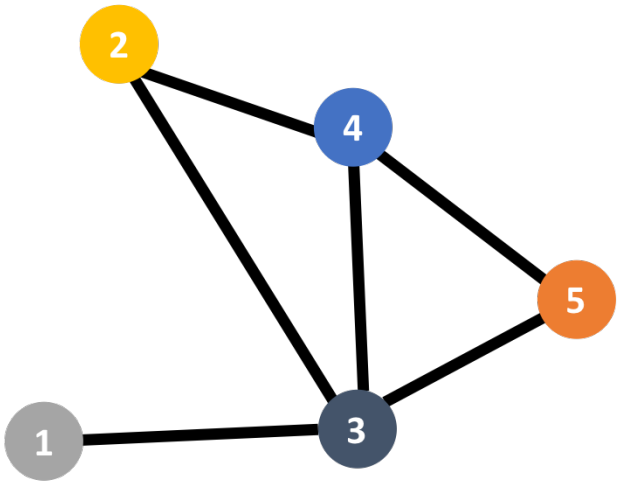


Returns exact solution, but requires n matrix-vector multiplies.

We will see how to use $m \ll n$ multiplies by using randomness and allowing for small approximation error.

Motivating Example

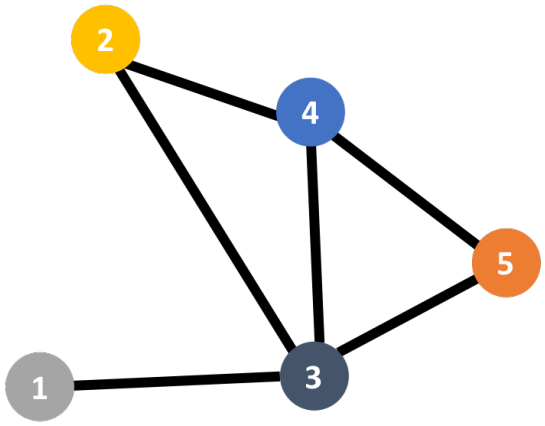
The number of triangles or other small ‘motifs’ is an important metric of network connectivity. E.g., important in computing the network clustering coefficient



How long does it take to exactly compute the number of triangles in the graph?

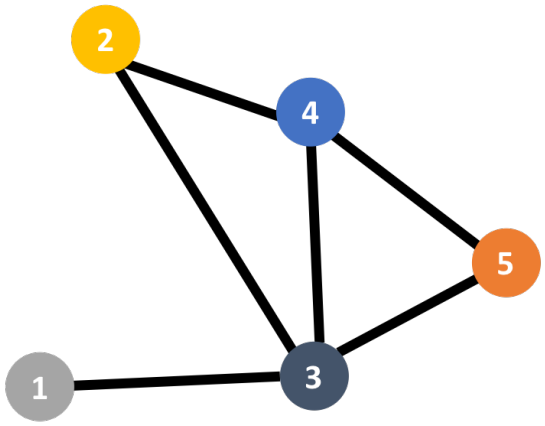
Motivating Example

Can use the adjacency matrix $B \in \{0,1\}^{n \times n}$ to write the number of triangles in a linear algebraic way.



B

0	0	1	0	0
0	0	1	1	0
1	1	0	1	1
0	1	1	0	1
0	0	1	1	0

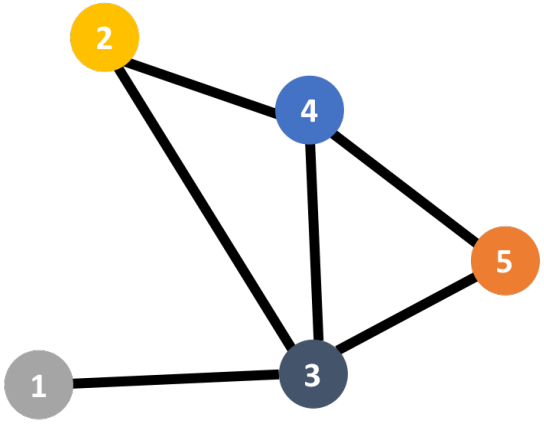


- B_{ij} indicates the number of 1-step paths (edges) from i, j
- $[B^2]_{ij}$ indicates the number of 2-step paths from i, j
- $[B^3]_{ij}$ indicates the number of 3-step paths from i, j

B_{ii} is the number of length 3-paths from i back to i. Thus,

$$\frac{1}{6}\,\mathrm{tr}(B^3) = \#\text{ triangles.}$$

Motivating Example



B^3

0	1	4	2	1
1	2	6	5	2
4	6	4	6	6
2	5	6	4	5
1	2	6	5	2

$$\frac{1}{6}\,\mathbf{tr}(B^3) = \#\text{ triangles.}$$

- Explicitly forming B^3 and computing $\mathbf{tr}(B^3)$ takes $O(n^3)$ time.
- Can multiply B^3 by a vector in $3 \cdot |E| = O(n^2)$ operations.
- So a trace estimation algorithm using m queries, yields an $O(m \cdot |E|)$ time approximate triangle counting algorithm.