COMPSCI 614: Randomized Algorithms with Applications to Data Science

Prof. Cameron Musco

University of Massachusetts Amherst. Spring 2024.

Lecture 12

Logistics

- The midterm is the Thursday after break in class.
- I will hold a review session Monday from 3-4:30pm and Tuesday in class.
- There is no real quiz this week, but see Weekly Quizzes section on Moodle for a single question quiz where you can mark if you attended Sally Dong's job talk for extra credit.

Summary

Last Time:

- Finish up fast low-rank approximation via approximate matrix multiplication.
- Start on stochastic trace estimation and motivation for matrix-vector query algorithms.

Today:

- Finish stochastic trace estimation.
- Hutchinson's estimator and full analysis.

Matrix Trace

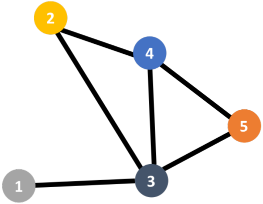
The trace of a matrix $A \in \mathbb{R}^{n\times n}$ is the sum of it diagonal entries.

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

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

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

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

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

- Explicitly forming B^3 and computing $\mathrm{tr}(B^3)$ 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.

Other Examples

Example 2: Hessian/Jacobian matrix-vector products.

- For vector x, $\nabla \mathbf{f}(y)x$ and $\nabla^2 f(y)x$ can often be computed efficiently using finite difference methods or explicit differentiation (e.g., via backpropagation).
- Do not need to fully form $\nabla f(y)$ or $\nabla^2 f(y)$.
- Many applications of estimating the traces of these matrices, e.g., in analyzing neural network convergence, in optimization of score-based methods, etc.
- $\text{tr}(\nabla^2 f(y)x)$: Laplacian
- $\text{tr}(\nabla f(y)x)$: Divergence

Other Examples

Example 3: A is a function of another (explicit) matrix B, $A = f(B)$ that can be applied efficiently via an iterative method.

- Repeated multiplication to apply $A = B^3$.
- Conjugate gradient, MINRES, or any linear system solver:

$$A = B^{-1}.$$

- Lanczos method, polynomial/rational approximation:

$$A = \exp(B),\ A = \sqrt{B},\ A = \log(B),\ \text{etc.}$$

- These methods run in $n^2 \cdot C$ time, where C depends on properties of B. Typically $C \ll n$ so $n^2 \cdot C \ll n^3$.

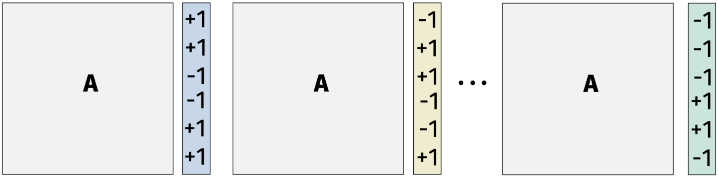
Matrix Function Examples

- Log-likelihood computation in Bayesian optimization, experimental design. $\text{tr}(\log(B)) = \log\det(B)$.
- Estrada index, a measure of protein folding degree and more generally, network connectivity. $\text{tr}(\exp(B))$.
- Trace inverse, which is important in uncertainty quantification and many other scientific computing applications. $\text{tr}(B^{-1})$
- Information about the matrix eigenvalue spectrum, since $\text{tr}(f(B)) = \sum_{i=1}^{n} f(\lambda_i)$, where λ_i is B's i^{th} eigenvalue.
- E.g., counting the number of eigenvalues in an interval, spectral density estimation, matrix norms
- See e.g., [Ubaru, and Saad 2017].

Hutchinson's Method

Hutchinson 1991, Girard 1987:

- Draw $\mathbf{x}_1, \ldots, \mathbf{x}_m \in \mathbb{R}^n$ i.i.d. with random $\{+1, -1\}$ entries.
- Return $\bar{\mathsf{T}} = \frac{1}{m}\sum_{i=1}^{m} \mathbf{x}_i^T A \mathbf{x}_i$ as an approximation to $\operatorname{tr}(A)$.



- One of the earliest examples I know of a randomized algorithm for linear algebraic computation.

Hutchinson's Method Error Bound

Theorem

Let $\overline{\mathsf{T}}$ be the trace estimate returned by Hutchinson's method. If $m = O\left(\frac{1}{\delta\epsilon^2}\right)$, then with probability $\geq 1 - \delta$,

$$\left|\overline{\mathsf{T}} - \mathrm{tr}(A)\right| \leq \epsilon \|A\|_F$$

If A is symmetric positive semidefinite (PSD) then

$$\|A\|_F = \sqrt{\sum_{i=1}^{n} \lambda_i^2} \leq \sum_{i=1}^{n} \lambda_i = \mathrm{tr}(A).$$

So for PSD A: $(1-\epsilon)\,\mathrm{tr}(A) \leq \overline{\mathsf{T}} \leq (1+\epsilon)\,\mathrm{tr}(A).$

Proof Approach

Theorem

Let $\overline{\mathsf{T}}$ be the trace estimate returned by Hutchinson's method. If $m = O\left(\frac{1}{\delta\epsilon^2}\right)$, then with probability $\geq 1-\delta$,

$$\left|\overline{\mathsf{T}} - \operatorname{tr}(A)\right| \leq \epsilon\|A\|_F$$

1. Show that $\mathbb{E}[\overline{\mathsf{T}}] = \operatorname{tr}(A)$.
2. Bound $\operatorname{Var}[\overline{\mathsf{T}}]$.
3. Apply Chebyshev's inequality.

A tighter proof that uses the Hanson-Wright inequality, an exponential concentration inequality for quadratic forms, can improve the δ dependence to $\log(1/\delta)$ – we'll cover this later in the class.

Expectation Analysis

Hutchinson's Estimator::

- Draw $\mathbf{x}_1, \ldots, \mathbf{x}_m \in \mathbb{R}^n$ i.i.d. with random $\{+1, -1\}$ entries.
- Return $\bar{\mathrm{T}} = \frac{1}{m}\sum_{i=1}^m \mathbf{x}_i^T A \mathbf{x}_i$ as an approximation to $\mathrm{tr}(A)$.

By linearity of expectation, $\mathbb{E}[\bar{\mathrm{T}}] = \mathbb{E}[\mathbf{x}^T A \mathbf{x}]$ for a single random ± 1 vector $\mathbf{x}$.

$$\mathbb{E}[\mathbf{x}^T A \mathbf{x}] = \mathbb{E}\sum_{i=1}^n \sum_{j=1}^n \mathbf{x}_i \mathbf{x}_j A_{ij} = \sum_{i=1}^n \sum_{j=1}^n A_{ij} \cdot \mathbb{E}[\mathbf{x}_i \mathbf{x}_j] = \sum_{i=1}^n A_{ii}.$$

- When $i \neq j$, $\mathbf{x}_i \mathbf{x}_j = 1$ with probability $1/2$ and -1 with probability $1/2$, so $\mathbb{E}[\mathbf{x}_i \mathbf{x}_j] = 0$. When $i = j$, $\mathbf{x}_i \mathbf{x}_j = 1$, so $\mathbb{E}[\mathbf{x}_i \mathbf{x}_j] = 1$.
- So the estimator is correct in expectation: $\mathbb{E}[\bar{\mathrm{T}}] = \mathrm{tr}(A)$.

Variance Bound

Hutchinson's Estimator::

- Draw $\mathbf{x}_1, \ldots, \mathbf{x}_m \in \mathbb{R}^n$ i.i.d. with random $\{+1, -1\}$ entries.
- Return $\bar{\mathsf{T}} = \frac{1}{m} \sum_{i=1}^m \mathbf{x}_i^T A \mathbf{x}_i$ as an approximation to $\mathrm{tr}(A)$.

$$\mathrm{Var}[\bar{\mathsf{T}}] = \frac{1}{m} \mathrm{Var}[\mathbf{x}^T A \mathbf{x}] = \frac{1}{m} \mathrm{Var}\left[\sum_{i=1}^n \sum_{j=1}^n \mathbf{x}_i \mathbf{x}_j A_{ij}\right]$$

Can we apply linearity of variance here? Almost – need to remove repeated terms, and then can use pairwise independence.

$$\mathrm{Var}[\bar{\mathsf{T}}] = \frac{1}{m} \mathrm{Var}\left[\sum_{i=1}^n A_{ii} + \sum_{i=1}^n \sum_{j>i} \mathbf{x}_i \mathbf{x}_j (A_{ij} + A_{ji})\right]$$

$$= \frac{1}{m} \sum_{i=1}^n \sum_{j>i} \mathrm{Var}[\mathbf{x}_i \mathbf{x}_j] \cdot (A_{ij} + A_{ji})^2 \leq \frac{1}{m} \sum_{i=1}^n \sum_{j>i} 2A_{ij}^2 + 2A_{ji}^2 \leq \frac{2\|A\|_F^2}{m}.$$

Final Analysis

Hutchinson's Estimator::

- Draw $\mathbf{x}_1, \ldots, \mathbf{x}_m \in \mathbb{R}^n$ i.i.d. with random $\{+1, -1\}$ entries.
- Return $\bar{\mathsf{T}} = \frac{1}{m}\sum_{i=1}^{m} \mathbf{x}_i^T A \mathbf{x}_i$ as an approximation to $\mathrm{tr}(A)$.

Chebyshev's inequality implies that, for $m = \frac{2}{\delta\epsilon^2}$:

$$\Pr\left[\left|\bar{\mathsf{T}} - \mathrm{tr}(A)\right| \geq \epsilon\|A\|_F\right] \leq \frac{2\|A\|_F^2/m}{\epsilon^2\|A\|_F^2} = \delta.$$

Could we have gotten a better bound by applying Bernstein's inequality to $\sum_{i=1}^{n}\sum_{j>i} \mathbf{x}_i\mathbf{x}_j(A_{ij} + A_{ji})$?

Hanson-Wright is an exponential concentration bound that can be used in the specific case – improves bound to $m = O\left(\frac{\log(1/\delta)}{\epsilon^2}\right)$.

Optimality of Hutchinson's Method

The $m = O\left(\frac{\log(1/\delta)}{\epsilon^2}\right)$ bound given by the Hanson-Wright inequality is tight.

- Any algorithm that only uses queries of the form $\mathbf{x}_i^T A \mathbf{x}_i$ requires $\Omega\left(\frac{\log(1/\delta)}{\epsilon^2}\right)$ samples to estimate $\text{tr}(A)$ to error $\pm\epsilon\,\text{tr}(A)$ for PSD A [Wimmer, Wu, Zhang 2014].
- We recently showed that using the full power of matrix-vector queries, one can achieve $O\left(\frac{\log(1/\delta)}{\epsilon}\right)$ queries for PSD matrices.