TRACE ESTIMATION AS AN ALGORITHMIC PRIMITIVE

CONNER DIPAOLO*

Abstract. This lecture note details motivation, results, and upper/lower sample complexity bounds for estimating the trace of large matrices when computing elements of $\boldsymbol{A}$ is about as slow as computing $\langle \boldsymbol{A}\boldsymbol{x}, \boldsymbol{x}\rangle$ for vectors $\boldsymbol{x}$.

1. Motivation: Computing Schatten Norms. Let $\boldsymbol{A}$ be any positive definite matrix in M_n. We're going to assume that $\boldsymbol{A}$ is real but these results largely carry over directly to the complex case. This matrix might represent the error of a low rank matrix factorization, or might specify something about the curvature of the loss surface of a deep neural network. In both cases, knowing how 'large' $\boldsymbol{A}$ is, in some norm, would help us determing how good the factorization is or how drastically our loss surface is curving.

For whatever reason, let's suppose we want to measure $\boldsymbol{A}$ in a Schatten-p norm, where $p = 1, 2, \ldots < \infty$ is an integer.

DEFINITION 1.1. *Let $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_n$ be the singular values of $\boldsymbol{A}$ (which are the same as the eigenvalues since $\boldsymbol{A}$ is positive definite). The Schatten-p norm of $\boldsymbol{A}$ is*

$$\|\boldsymbol{A}\|_p^p = \sum_{i=1}^{n} \lambda_1^p = \operatorname{tr}(\boldsymbol{A}^p).$$

The normal procedure for computing $\|\boldsymbol{A}\|_p$ goes as follows.

Algorithm 1.1 Naive Schatten norm computation

Data: A positive semi-definite matrix $\boldsymbol{A} \in \mathsf{M}_n$ and an integer $p = 1, 2, \ldots < \infty$.
Result: The Schatten-p norm $\|A\|_p$ using $\frac{4}{3}n^3 + np + 17 = \mathcal{O}(n^3 + np)$ flops.
Overwrite $\boldsymbol{A}$ with $\operatorname{diag}(\lambda_1, \ldots, \lambda_n)$ using $\sim 4n^3/3$ flops. (Golub and Van Loan, 2012, Alg 8.3.3)
Compute $\boldsymbol{A} \leftarrow \boldsymbol{A}^p$ using $n(p-1)$ flops.
Compute $\operatorname{tr} \boldsymbol{A} = \sum_{i=1}^{n} \boldsymbol{A}_{ii}$ using $n-1$ flops.
Return $\|\boldsymbol{A}\|_p^p = (\operatorname{tr} \boldsymbol{A})^{1/p}$ using about 18 flops.

But what happens when n is large, say $n \geq 10,000$? Then the n^3 computation time can become unbearable because the eigendecomposition becomes slow. Indeed, on my 1.4GHz computer the estimated computation time for arbitrary $\boldsymbol{A}$ is at least a day for $n = 50,000$ and $p = 2$. What can we do to speed this up? In particular, how can we drop the n^3 dependence?

Well, one way to do this is to note that for any $\boldsymbol{x}$, we can compute $\boldsymbol{A}\boldsymbol{x}$ in $2n^2 - n$ flops, and as a result we can compute $\boldsymbol{A}^p\boldsymbol{x}$ in $p(2n^2 - n) = 2n^2p - np$ flops. In particular, we can compute $\langle \boldsymbol{A}^p\boldsymbol{x}, \boldsymbol{x}\rangle$ in $2n^2p - np + 2n - 1$ flops. Now

$$\|\boldsymbol{A}\|_p^p = \operatorname{tr} \boldsymbol{A}^p = \sum_{i=1}^{n} \boldsymbol{A}_{ii}^p = \sum_{i=1}^{n} \langle \boldsymbol{A}^p \boldsymbol{e}_i, \boldsymbol{e}_i\rangle,$$

so naive computation in this form gives an algorithm for computing the Schatten-p norm in $n(2n^2p - np + 2n - 1) = 2n^3p - n^2p + 2n^2 - n + 18 = \mathcal{O}(n^3p)$ flops. Immediately, this performs worse than our original Algorithm 1.1, but this provides an interesting interpretation of $\|\boldsymbol{A}\|_p^p$ as an expectation.

$$\|\boldsymbol{A}\|_p^p = n\mathbf{E}\langle \boldsymbol{A}^p \boldsymbol{e}_i, \boldsymbol{e}_i\rangle, \text{ where } i \sim \text{Uniform}\{1, 2, \ldots, n\}.$$

If we take a hint from Monte-Carlo methods (as well as ideas from the world of concentration inequalities), this means that the following randomized algorithm should return a good *estimate* for $\|\boldsymbol{A}\|_p$ with high probability, while requiring significantly fewer flops for large n.

*Harvey Mudd College (cdipaolo@hmc.edu)

Algorithm 1.2 Naive randomized Schatten norm computation

Data: Positive semi-definite $\boldsymbol{A} \in \mathsf{M}_n$, integer $p = 1, 2, \ldots < \infty$, and accuracy parameter $m \in \{1, 2, \ldots\}$
Result: Estimated Schatten-p norm $S \approx \|\boldsymbol{A}\|_p$ using $2mn^2p - mnp + m + 19 = \mathcal{O}(mn^2p)$ flops.
Initialize $S \leftarrow 0$.
Initialize $\boldsymbol{x} \leftarrow 0 \in \mathbb{C}^n$.
for $1 \le k \le m$ **do**
  Sample an index $i \sim \text{Uniform}\{1, 2, \ldots, n\}$.
  Let $\boldsymbol{x} \leftarrow \boldsymbol{e}_i$.
  for $1 \le j \le p$ **do**
    Compute $\boldsymbol{x} \leftarrow \boldsymbol{A}\boldsymbol{x}$ using $2n^2 - n$
  end
  Accumulate $S \leftarrow S + \boldsymbol{x}_i$ using 1 flop.
end
Let $S \leftarrow \frac{n}{m} S$ using 1 flop.
Return $\|\boldsymbol{A}\|_p \approx S^{1/p}$ using about 18 flops.

Actual bounds on how good of an estimate Algorithm 1.2 returns will follow from results in the later sections, and are provided explicitly in (Woodruff et al., 2014, Thm 69), but for now just remark how the simple realization that the trace is an expected inner product

$$\operatorname{tr}(\boldsymbol{A}^p) = n\mathbf{E}\langle \boldsymbol{A}^p \boldsymbol{e}_i, \boldsymbol{e}_i\rangle = \mathbf{E}\langle \boldsymbol{A}^p \sqrt{n}\boldsymbol{e}_i, \sqrt{n}\boldsymbol{e}_i\rangle, \text{ where } i \sim \text{Uniform}\{1, 2, \ldots, n\}.$$

allowed us to drop an n^3 dependence in our algorithm to a randomized algorithm with n^2 dependence without bringing in any heavy machinery. One thing we can say without any real work is that Algorithm 1.2 returns an estimate $S_m \to \|\boldsymbol{A}\|_p$ as $m \to \infty$ by the law of large numbers. (Wasserman, 2013, Thm 5.18) This follows since by positive definiteness $\mu = \mathbf{E}|\langle \boldsymbol{A}^p \sqrt{n}\boldsymbol{e}_i, \sqrt{n}\boldsymbol{e}_i\rangle| = \mathbf{E}\langle \boldsymbol{A}^p \sqrt{n}\boldsymbol{e}_i, \sqrt{n}\boldsymbol{e}_i\rangle = \operatorname{tr}(\boldsymbol{A}^p) < \infty$.

1.1. Trace Estimation as an Algorithmic Primitive. The above progress for Schatten norm estimation made progress by representing our quantity of interest as a trace, and then applying randomized trace estimators to give us our quantity of interest. This idea has become a canonical tool for computing traces of functions of matrices represented by power series. Indeed, if we want to compute some function $\operatorname{tr}(f(\boldsymbol{A}))$ where

$$f(x) = \sum_{n=0}^{\infty} a_n x^n$$

is analytic, then

$$\operatorname{tr}\big(f(\boldsymbol{A})\big) = \sum_{n=0}^{\infty} a_n \operatorname{tr}(\boldsymbol{A}^n) \approx \sum_{n=0}^{N} a_n \operatorname{tr}(\boldsymbol{A}^n)$$

for some small N depending on f and $\boldsymbol{A}$. From here we can approximate further each $\operatorname{tr}(\boldsymbol{A}^n)$ by a trace estimator to compute an approximation to the original quantity of interest $\operatorname{tr}(f(\boldsymbol{A}))$. This exact idea has been used to create fast input-sparsity time algorithms for computing

$$\log\det(\boldsymbol{I} + \boldsymbol{A}) = \operatorname{tr}\log(\boldsymbol{I} + \boldsymbol{A}) = \sum_{n=1}^{\infty} \frac{(-1)^{n+1}}{n} \operatorname{tr}(\boldsymbol{A}^n)$$

in the recent work of Boutsidis et al. (2017). Trace estimation has been used as a primitive for creating better algorithms for counting the number of triangles in large graphs, (Avron, 2010) counting the number of eigenvalues of a matrix inside an interval, (Di Napoli et al., 2016) and even estimating the entire spectrum of matrices. (Adams et al., 2018) In the coming sections, we will go into detail on this primitive, why it works, and why we can't hope to do better by other means.

1.2. Computing Traces. Trace computation is easy if we have full access to a matrix $\boldsymbol{A}$. Indeed,

$$\operatorname{tr}\boldsymbol{A} = \sum_{i=1}^{n} \boldsymbol{A}_{ii}$$

can be computed in $n-1$ flops. But what if we don't have direct access to $\boldsymbol{A}$, or in particular if computing $\boldsymbol{A}_{ii}$ takes about as long as computing $\boldsymbol{x}^*\boldsymbol{A}\boldsymbol{x}$ for any input vector $\boldsymbol{x}$? The above motivation suggests a natural algorithmic process for computing the trace of matrices of this type. We start with some random vector $\boldsymbol{r}$ distributed so that

$$\mathbf{E}\langle \boldsymbol{A}\boldsymbol{r}, \boldsymbol{r}\rangle = \operatorname{tr} \boldsymbol{A}$$

for all positive matrices $\boldsymbol{A} \in \mathsf{M}_n$. From then, we sample m identical and independent copies of $\boldsymbol{r}$: $\boldsymbol{r}_i$ for $i = 1, 2, \dots, m$. Finally, we return

$$T_m = \frac{1}{m}\sum_{i=1}^{n} \langle \boldsymbol{A}\boldsymbol{r}_i, \boldsymbol{r}_i\rangle \approx \operatorname{tr} \boldsymbol{A}$$

For which random vectors $\boldsymbol{r}$ does this informal algorithm work? Well, it must be that $\mathbf{E}\langle \boldsymbol{A}\boldsymbol{r}, \boldsymbol{r}\rangle = \operatorname{tr} \boldsymbol{A}$, so by the previous section we know $\boldsymbol{r} \sim \text{Uniform}\{\sqrt{n}\boldsymbol{e}_i\}$ works. It turns out that a much larger class of vectors works, however.

THEOREM 1.2. $\mathbf{E}\langle \boldsymbol{A}\boldsymbol{r}, \boldsymbol{r}\rangle = \operatorname{tr} \boldsymbol{A}$ *for all positive definite* $\boldsymbol{A} \in \mathsf{M}_n$ *if and only if* $\mathbf{E}\boldsymbol{r}\boldsymbol{r}^* = \mathbf{E}\boldsymbol{r} \otimes \boldsymbol{r} = \boldsymbol{I}$.

Proof. By linearity and the cyclic properties of the trace,

$$\mathbf{E}\langle \boldsymbol{A}\boldsymbol{r}, \boldsymbol{r}\rangle = \mathbf{E} \operatorname{tr}(\boldsymbol{r}^*\boldsymbol{A}\boldsymbol{r}) = \mathbf{E} \operatorname{tr}(\boldsymbol{A}\boldsymbol{r}\boldsymbol{r}^*) = \operatorname{tr}\big(\boldsymbol{A}(\mathbf{E}\boldsymbol{r}\boldsymbol{r}^*)\big)$$

Let $\boldsymbol{\Sigma} = \mathbf{E}\boldsymbol{r}\boldsymbol{r}^*$. If $\boldsymbol{\Sigma} = \boldsymbol{I}$ then clearly $\mathbf{E}\langle \boldsymbol{A}\boldsymbol{r}, \boldsymbol{r}\rangle = \operatorname{tr}(\boldsymbol{A}\boldsymbol{\Sigma}) = \operatorname{tr}(\boldsymbol{A})$ for all $\boldsymbol{A} \in \mathsf{M}_n$. On the other hand, if

$$\operatorname{tr}(\boldsymbol{A}\boldsymbol{\Sigma}) = \operatorname{tr}(\boldsymbol{A})$$

for all positive *semi*-definite $\boldsymbol{A} \in \mathsf{M}_n$ then in particular

$$\boldsymbol{\Sigma}_{ii} = \boldsymbol{e}_i^*\boldsymbol{\Sigma}\boldsymbol{e}_i = \operatorname{tr}(\boldsymbol{e}_i\boldsymbol{e}_i^*\boldsymbol{\Sigma}) = \operatorname{tr}(\boldsymbol{e}_i\boldsymbol{e}_i^*) = 1.$$

But then

$$2 + 2\boldsymbol{\Sigma}_{ij} = \boldsymbol{\Sigma}_{ii} + \boldsymbol{\Sigma}_{jj} + 2\boldsymbol{\Sigma}_{ij} = \operatorname{tr}\big((\boldsymbol{e}_i + \boldsymbol{e}_j)(\boldsymbol{e}_i + \boldsymbol{e}_j)^*\boldsymbol{\Sigma}\big) = \operatorname{tr}(\boldsymbol{e}_i + \boldsymbol{e}_j)(\boldsymbol{e}_i + \boldsymbol{e}_j)^* = 2$$

by the same logic so $\boldsymbol{\Sigma}_{ij} = 0$ for $i \neq j$. In sum $\boldsymbol{\Sigma} = \boldsymbol{I}$ if $\mathbf{E}\langle \boldsymbol{A}\boldsymbol{r}, \boldsymbol{r}\rangle = \operatorname{tr} \boldsymbol{A}$ for all positive *semi*-definite $\boldsymbol{A}$. But by constructing positive semi-definite matrices as a limit of strictly positive definite matrices (e.g. by adding $\epsilon\boldsymbol{I}$) this also holds if $\mathbf{E}\langle \boldsymbol{A}\boldsymbol{r}, \boldsymbol{r}\rangle = \operatorname{tr} \boldsymbol{A}$ for only the strictly positive definite matrices. ∎

This implies that standard normal vectors $\boldsymbol{r} \sim \mathcal{N}(0, \boldsymbol{I})$ as well as Rademacher vectors $\boldsymbol{r} \sim \text{Uniform}\{-1, 1\}^n$ both work for constructing trace estimators.

2. Upper Bounds on Performance of the Hutchinson Estimator. The above work suggests that if $\boldsymbol{r}_i \sim \text{Uniform}\{-1, 1\}^n$ are identical and independent Rademacher random vectors then

$$H_m = \frac{1}{m}\sum_{i=1}^{m} \langle \boldsymbol{A}\boldsymbol{r}_i, \boldsymbol{r}_i\rangle \to \operatorname{tr} \boldsymbol{A}$$

almost surely for any fixed matrix $\boldsymbol{A}$ are $m \to \infty$. We can get a more quantitative bound on the performance if $\boldsymbol{A}$ is known to be positive semi-definite, though.

THEOREM 2.1 (Roosta-Khorasani and Ascher Thm 1). *If* $\boldsymbol{A} \in \mathsf{M}_n$ *is positive semi-definite, then*

$$\mathbf{P}\big(|H_m - \operatorname{tr} \boldsymbol{A}| > \epsilon \operatorname{tr} \boldsymbol{A}\big) < 2e^{-\frac{m}{2}\left(\frac{\epsilon^2}{2} - \frac{\epsilon^3}{3}\right)}.$$

In other words, if $m \geq \frac{12 \log(\frac{2}{\delta})}{\epsilon^2(3-2\epsilon)}$ *then the relative error* $|H_m - \operatorname{tr} \boldsymbol{A}| \leq \epsilon \operatorname{tr} \boldsymbol{A}$ *with probability at least* $1 - \delta$. *Moreover, if* $\epsilon < \frac{1}{2}$, *then* $m \geq \frac{6}{\epsilon^2} \log \frac{2}{\delta}$ *implies the relative error* $|H_m - \operatorname{tr} \boldsymbol{A}| \leq \epsilon \operatorname{tr} \boldsymbol{A}$ *with probability at least* $1 - \delta$.

Note crucially that the necessary number of queries m needed to construct an ϵ-relative error approximation to $\operatorname{tr} \boldsymbol{A}$ with high probability doesn't depend on the dimensions of $\boldsymbol{A}$, which is why this approach lends itself so well to reducing dimension dependence of computational algorithms.

The proof of Theorem 2.1 is surprisingly tedious, and crucially relies on a sequence of results from Achlioptas (2001) which bound the moment generating function of the square of a projection of a Rademacher by reducing to a Gaussian case. To avoid these troubles, we will prove the identical result for an analogous estimator

$$G_n = \frac{1}{m} \sum_{i=1}^{m} \langle \boldsymbol{A}\boldsymbol{g}_i, \boldsymbol{g}_i \rangle$$

where $\boldsymbol{g}_i \sim \mathcal{N}(0, I)$ identically and independently. Actually sampling Gaussian vectors and computing G_n takes a bit more work than G_n, but from a statistical perspective these work just as well. Note that the result presented here is stronger than that presented as (Roosta-Khorasani and Ascher, 2015, Thm 3), and brings the current best known sample complexity for trace estimators of Hutchinson type to a tie between the Gaussian and Rademacher cases.

THEOREM 2.2. *If $\boldsymbol{A} \in \mathsf{M}_n$ is positive semi-definite, then*

$$\mathbf{P}\big(|G_m - \operatorname{tr} \boldsymbol{A}| > \epsilon \operatorname{tr} \boldsymbol{A}\big) < 2e^{-\frac{m}{2}\left(\frac{\epsilon^2}{2} - \frac{\epsilon^3}{3}\right)}.$$

In other words, if $m \geq \frac{12 \log(\frac{2}{\delta})}{\epsilon^2(3-2\epsilon)}$ then the relative error $|G_m - \operatorname{tr} \boldsymbol{A}| \leq \epsilon \operatorname{tr} \boldsymbol{A}$ with probability at least $1 - \delta$. Moreover, if $\epsilon < \frac{1}{2}$, then $m \geq \frac{6}{\epsilon^2} \log \frac{2}{\delta}$ implies the relative error $|G_m - \operatorname{tr} \boldsymbol{A}| \leq \epsilon \operatorname{tr} \boldsymbol{A}$ with probability at least $1 - \delta$.

Proof. Diagonalize $\boldsymbol{A} = \boldsymbol{U}\boldsymbol{\Lambda}\boldsymbol{U}^*$ where $\boldsymbol{U}$ is unitary and $\boldsymbol{\Lambda}$ is diagonal. Let $\boldsymbol{z}_i = \boldsymbol{U}^* \boldsymbol{g}_i$ so that

$$\begin{aligned}
\mathbf{P}\big(G_m > (1+\epsilon) \operatorname{tr} \boldsymbol{A}\big) &= \mathbf{P}\left(\sum_{i=1}^{m} \boldsymbol{g}_i^* \boldsymbol{A}\boldsymbol{g}_i > (1+\epsilon) \operatorname{tr} \boldsymbol{A}\right) \\
&= \mathbf{P}\left(\sum_{i=1}^{m} \boldsymbol{z}_i^* \boldsymbol{\Lambda}\boldsymbol{z}_i > m(1+\epsilon) \operatorname{tr} \boldsymbol{A}\right) \\
&= \mathbf{P}\left(\sum_{i=1}^{m} \sum_{j=1}^{n} \lambda_j \boldsymbol{z}_{ij}^2 > m(1+\epsilon) \operatorname{tr} \boldsymbol{A}\right) \\
&= \mathbf{P}\left(\sum_{j=1}^{n} \frac{\lambda_j}{\operatorname{tr} \boldsymbol{A}} \sum_{i=1}^{m} \boldsymbol{z}_{ij}^2 > m(1+\epsilon)\right) \\
&= \mathbf{P}\left(\exp\left(t \sum_{j=1}^{n} \frac{\lambda_j}{\operatorname{tr} \boldsymbol{A}} \sum_{i=1}^{m} \boldsymbol{z}_{ij}^2\right) > e^{(1+\epsilon)mt}\right) \\
&\leq e^{-(1+\epsilon)mt} \, \mathbf{E} \exp\left(\sum_{j=1}^{n} \frac{\lambda_j}{\operatorname{tr} \boldsymbol{A}} \sum_{i=1}^{m} t\boldsymbol{z}_{ij}^2\right)
\end{aligned}$$

for $t > 0$ by Markov's inequality. (Wasserman, 2013, Thm 4.1) Now by convexity of the exponential and linearity of expectation

$$\mathbf{E} \exp\left(\sum_{j=1}^{n} \frac{\lambda_j}{\operatorname{tr} \boldsymbol{A}} \sum_{i=1}^{m} t\boldsymbol{z}_{ij}^2\right) \leq \sum_{j=1}^{n} \frac{\lambda_j}{\operatorname{tr} \boldsymbol{A}} \mathbf{E} \exp\left(\sum_{i=1}^{m} t\boldsymbol{z}_{ij}^2\right) = \sum_{j=1}^{n} \frac{\lambda_j}{\operatorname{tr} \boldsymbol{A}} \prod_{i=1}^{m} \mathbf{E} e^{t\boldsymbol{z}_{ij}^2}$$

since $\lambda_i \geq 0$ and $\sum_{i=1}^{n} \lambda_i = \operatorname{tr} \boldsymbol{A}$, and for a fixed j, $\boldsymbol{z}_{ij}$ are independent and identically distributed. Now, to bound the moment generating function, observe that $\boldsymbol{z}_i$ are just independent standard normal vectors by

the rotation invariance of the Gaussian and unitary nature of $\boldsymbol{U}$. This implies that $\boldsymbol{z}_{ij} \sim \mathcal{N}(0,1)$ and hence $\boldsymbol{z}_{ij}^2$ is just a χ^2 random variable with one degree of freedom. We then know (see Wasserman (2013)) that

$$\mathbf{E} e^{t\boldsymbol{z}_{ij}^2} = \frac{1}{\sqrt{1-2t}}$$

so long as $0 < t < 1/2$. This implies that

$$\mathbf{E} \exp\left(\sum_{j=1}^n \frac{\lambda_j}{\operatorname{tr} \boldsymbol{A}} \sum_{i=1}^m t\boldsymbol{z}_{ij}^2\right) \leq \left(\frac{1}{\sqrt{1-2t}}\right)^m$$

for these $0 < t < 1/2$ and hence taking $t = \frac{1}{2}\frac{\epsilon}{1+\epsilon} < 1/2$ we have

$$\mathbf{P}\big(G_m > (1+\epsilon)\operatorname{tr} \boldsymbol{A}\big) \leq \left(\frac{1}{\sqrt{1-2t}}\right)^m e^{-(1+\epsilon)mt} = \left((1+\epsilon)e^{-\epsilon}\right)^{m/2} < e^{-\frac{m}{2}\left(\frac{\epsilon^2}{2} - \frac{\epsilon^3}{3}\right)},$$

relying on the fact that $(1+\epsilon)e^{-\epsilon} < e^{\frac{\epsilon^3}{3} - \frac{\epsilon^2}{2}}$ for all $\epsilon > 0$, verifiable by simple calculus. Essentially the same argument for the lower tail tells us via a union bound that

$$\mathbf{P}(|G_m - \operatorname{tr} \boldsymbol{A}| > \epsilon \operatorname{tr} \boldsymbol{A}) < 2e^{-\frac{m}{2}\left(\frac{\epsilon^2}{2} - \frac{\epsilon^3}{3}\right)}. \qquad \square$$

3. Lower Bounds. We've illustrated extremely simple randomized algorithms for estimating the trace of a large matrix with implicit access. Moreover, we've shown upper bounds on these algorithms that guarantee fast performance with only $\frac{6}{\epsilon^2}\log\frac{2}{\delta}$ queries to $\boldsymbol{A}$. Nevertheless, questions still remain. Can we estimate traces with the same number of queries to $\boldsymbol{A}$ without using any randomness? Perhaps better estimators for the trace exist that only need $\frac{6}{\epsilon}\log\frac{2}{\delta}$ or $\frac{6}{\epsilon^2}\log\log\frac{2}{\delta}$ queries to $\boldsymbol{A}$. This section will illustrate why these conjectures are both false.

3.1. Deterministic Algorithms Can't Estimate Traces. Let's assume we have an oracle $\mathcal{O} : \boldsymbol{x} \mapsto \boldsymbol{A}\boldsymbol{x}$ and we'd like to deterministically estimate the trace of $\boldsymbol{A}$ by sequentially submitting queries m to $\mathcal{O}$, perhaps performing an uncomputable amount of work in between or rounds or after the querying process. The following theorem essentially says that the only real way to get an estimator T_n from this process with relative error

$$|T_m - \operatorname{tr} \boldsymbol{A}| < \operatorname{tr} \boldsymbol{A}$$

is to just compute

$$T_n = \sum_{i=1}^n \langle \mathcal{O}(\boldsymbol{e}_i), \boldsymbol{e}_i \rangle = \sum_{i=1}^n \boldsymbol{A}_{ii} = \operatorname{tr} \boldsymbol{A}$$

exactly.

THEOREM 3.1. *Any deterministic trace estimator T_m satisfying*

$$|T_m - \operatorname{tr} \boldsymbol{A}| \leq \epsilon \operatorname{tr} \boldsymbol{A}$$

for all positive semi-definite matrices $\boldsymbol{A} \in \mathsf{M}_n$ and some $0 < \epsilon < 1$ requires $m \geq n$.

Proof. We will follow via a resisting oracle. In particular, let's suppose that $\mathcal{O}$ returns $\mathcal{O}(\boldsymbol{x}_i) = 0$ for the first $n-1$ queries $\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_{n-1}$. At this point, both the zero matrix $\boldsymbol{A}_1 = 0$ and the orthogonal projection onto the orthogonal complement of the span of $\{\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_{n-1}\}$ $\boldsymbol{A}_2$ would return the 0 for these queries. If we output an estimator $T_m = T_{n-1}$ which satisfied

$$|T_m - \operatorname{tr} \boldsymbol{A}| \leq \epsilon \operatorname{tr} \boldsymbol{A}$$

for all positive semi-definite matrices $\boldsymbol{A} \in \mathsf{M}_n$, then since $\boldsymbol{A}_1$ is positive semi-definite this series of responses by $\mathcal{O}$ would ensure

$$|T_m| = |T_m - \operatorname{tr} \boldsymbol{A}_1| \leq \epsilon \operatorname{tr} \boldsymbol{A}_1 = 0,$$

so $T_m = 0$. On the other hand, the matrix $\boldsymbol{A}_2$ would give the same sequence of responses by $\mathcal{O}$ and so by determinism

$$\operatorname{tr} \boldsymbol{A}_2 = |\operatorname{tr} \boldsymbol{A}_2| = |T_m - \operatorname{tr} \boldsymbol{A}_2| \leq \epsilon \operatorname{tr} \boldsymbol{A}_2 < \operatorname{tr} \boldsymbol{A}_2,$$

a contradiction. It follows that at least n queries to $\mathcal{O}$ are needed to guarantee this uniform error bound deterministically. ∎

3.2. Simple Lower Bounds for Simple Randomized Trace Estimators. The next section will reference much more general lower bounds for randomized trace estimation. Those proofs are long and tedious. This section will present simple lower bounds due to the author and Weiqing Gu that guarantee the tightness of Roosta-Khorasani and Ascher (2015) without a ton of heavy machinery, at the expense of the full generality of the bounds of Wimmer et al. (2014).

Recall that the Hutchinson estimator of the trace of a matrix $\boldsymbol{A} \in \mathsf{M}_n$ is

$$H_m = \frac{1}{m} \sum_{i=1}^{m} \langle \boldsymbol{A}\boldsymbol{r}_i, \boldsymbol{r}_i \rangle$$

where $\boldsymbol{r}_i \sim \text{Uniform}\{-1, 1\}^n$ independently and identically. For this section, we will call H_n a Hutchinson-type estimator for the trace of a matrix $\boldsymbol{A}$ if

$$H_m = \frac{1}{m} \sum_{i=1}^{m} \langle \boldsymbol{A}\boldsymbol{r}_i, \boldsymbol{r}_i \rangle$$

where $\mathbf{E}\boldsymbol{r}_i\boldsymbol{r}_i^* = \boldsymbol{I}$ and $\boldsymbol{r}$ is real almost surely. Recall from Theorem 1.2 that $H_n \to \operatorname{tr} \boldsymbol{A}$ almost surely. While the original Hutchinson estimator returns the trace exactly in one query for diagonal matrices, it happens that every Hutchinson-type estimator has some matrix $\boldsymbol{A}$ for which $\mathbf{Var}(H_m) > 0$.

LEMMA 3.2. *If the real random vector $\boldsymbol{r}$ satisfies $\mathbf{E}\boldsymbol{r}\boldsymbol{r}^* = \boldsymbol{I}$, then $\mathbf{Var}(\langle \boldsymbol{A}\boldsymbol{r}, \boldsymbol{r} \rangle) > 0$ for some positive semi-definite $\boldsymbol{A} \in \mathsf{M}_n$.*

Proof. In light of Theorem 1.2, suppose to the contrary that $\langle \boldsymbol{A}\boldsymbol{r}, \boldsymbol{r} \rangle = \operatorname{tr} \boldsymbol{A}$ almost surely for every positive semi-definite $\boldsymbol{A} \in \mathsf{M}_n$. In particular,

$$\boldsymbol{r}_i^2 = \langle \boldsymbol{e}_i \boldsymbol{e}_i^* \boldsymbol{r}, \boldsymbol{r} \rangle = \operatorname{tr} \boldsymbol{e}_i \boldsymbol{e}_i^* = 1$$

almost surely for all $i, j = 1, 2, \ldots, n$, so $\boldsymbol{r} \in \{-1, 1\}^d$ almost surely. On the other hand, if $\boldsymbol{r}_i \neq \boldsymbol{r}_j$ with positive probability for some $i \neq j$ we would have

$$0 = (\boldsymbol{r}_i + \boldsymbol{r}_j)^2 = \langle (\boldsymbol{e}_i + \boldsymbol{e}_j)(\boldsymbol{e}_i + \boldsymbol{e}_j)^* \boldsymbol{r}, \boldsymbol{r} \rangle = \operatorname{tr}(\boldsymbol{e}_i + \boldsymbol{e}_j)(\boldsymbol{e}_i + \boldsymbol{e}_j)^* = 2$$

with positive probability, a contradiction, so almost surely every component of $\boldsymbol{r}$ is the same. But then $0 = \mathbf{E}\boldsymbol{r}_i\boldsymbol{r}_j = \mathbf{E}\boldsymbol{r}_i^2 = 1$, contradicting our assumption that $\mathbf{E}\boldsymbol{r}\boldsymbol{r}^* = \boldsymbol{I}$. ∎

The following theorem relies weakly on the above lemma to say that any Hutchinson-type estimator needs $\Omega(\frac{1}{\epsilon^2} \log \frac{1}{\delta})$ queries to $\boldsymbol{A}$ in order to achieve an ϵ-relative error approximation to $\operatorname{tr} \boldsymbol{A}$ with probability at least $1 - \delta$. Recall that W is the Lambert-W function, satisfying $W(x) = \log x - \log \log x + o(1) = \Theta(\log x)$ as $x \to \infty$. (Hoorfar and Hassani, 2007)

THEOREM 3.3. *Fix $0 < \delta < \frac{1}{10}$. For every Hutchinson-type estimator H_n there exists a positive semi-definite matrix $\boldsymbol{A} \in \mathsf{M}_n$ and an $\epsilon_0 > 0$ so that*

$$\mathbf{P}\big(|H_n - \operatorname{tr} \boldsymbol{A}| > \epsilon \operatorname{tr} \boldsymbol{A}\big) > \delta$$

whenever $0 < \epsilon < \epsilon_0$ and $n = \lfloor \frac{\mathbf{Var}(\langle \boldsymbol{A}\boldsymbol{x}, \boldsymbol{x} \rangle)}{\operatorname{tr}(\boldsymbol{A})^2} \frac{1}{\epsilon^2} W(\frac{2/\pi}{\delta^2}) \rfloor = \Theta(\frac{1}{\epsilon^2} \log \frac{1}{\delta})$.

Proof. Pick $\boldsymbol{A}$ so that $\mathbf{Var}(\langle \boldsymbol{A}\boldsymbol{r}, \boldsymbol{r} \rangle) > 0$ by the lemma. If $Z \sim \mathcal{N}(0, 1)$ is a standard normal random variable then

$$\mathbf{P}\big(|Z| > t\big) = 2\mathbf{P}(Z > t) > \sqrt{\frac{2}{\pi}} \frac{t}{t^2 + 1} e^{-t^2} \geq \frac{1}{\sqrt{2\pi}} \frac{1}{t} e^{-t^2}$$

when $t \geq 1$. (Cook, 2009) Setting the right hand side to 2δ and solving gives

$$\mathbf{P}\big(|Z| > 2^{-1/2}\sqrt{W(\pi^{-1}\delta^{-2})}\big) > 2\delta$$

whenever $2^{-1/2}\sqrt{W(\pi^{-1}\delta^{-2})} \geq 1$ or $0 < \delta \leq (e\sqrt{2\pi})^{-1}$. This is satisfied when $0 < \delta < \frac{1}{10}$. If we fix $n = \lfloor \frac{\mathbf{Var}(\langle \boldsymbol{A}\boldsymbol{x}, \boldsymbol{x}\rangle)}{\operatorname{tr}(\boldsymbol{A})^2} \frac{1}{\epsilon^2} W(\frac{2/\pi}{\delta^2}) \rfloor$ and write $\sigma^2 = \mathbf{Var}(\langle \boldsymbol{A}\boldsymbol{r}, \boldsymbol{r}\rangle) > 0$,

$$\mathbf{P}\big(|H_n - \operatorname{tr}\boldsymbol{A}| > \epsilon \operatorname{tr}\boldsymbol{A}\big) = \mathbf{P}\left(\frac{\sqrt{n}}{\sigma}|H_n - \operatorname{tr}\boldsymbol{A}| > \frac{\sqrt{n}}{\sigma}\epsilon \operatorname{tr}\boldsymbol{A}\right) \geq \mathbf{P}\left(\frac{\sqrt{n}}{\sigma}|H_n - \operatorname{tr}\boldsymbol{A}| > \frac{1}{\sqrt{2}}\sqrt{W(\pi^{-1}\delta^{-2})}\right).$$

Conveniently, the latter expression converges to $\mathbf{P}(|Z| \geq 2^{-1/2}\sqrt{W(\pi^{-1}\delta^{-2})})$ as $n \to \infty$, (Wasserman, 2013, Thm 5.10) and so

$$\lim_{\epsilon \to 0} \mathbf{P}\big(|H_n - \operatorname{tr}\boldsymbol{A}| > \epsilon \operatorname{tr}\boldsymbol{A}\big) > 2\delta.$$

This implies the existence of an $\epsilon_0 > 0$ so that $0 < \epsilon < \epsilon_0$ ensures $\mathbf{P}(|H_n - \operatorname{tr}\boldsymbol{A}| > \epsilon \operatorname{tr}\boldsymbol{A}) > \delta$ under our relation defining n. ∎

3.3. Generic Lower Bounds. Somewhat recent work by Wimmer et al. (2014) has shown that the upper bounds given by Roosta-Khorasani and Ascher (2015) are tight: no algorithm that can sequentially submit vectors $\boldsymbol{x}$ to an oracle $\mathcal{O} : \boldsymbol{x} \mapsto \langle \boldsymbol{A}\boldsymbol{x}, \boldsymbol{x}\rangle$ and possibly estimate the trace of $\boldsymbol{A}$ unless $\Omega(\frac{1}{\epsilon^2}\log\frac{1}{\delta})$ queries to $\boldsymbol{A}$ are made. In particular, Wimmer, Wu, and Zhang prove the following sequence of Theorems.

THEOREM 3.4 (Wimmer et al. Thm 1). *If we consider estimators for* $\operatorname{tr}\boldsymbol{A}$ *that pre-decide a distribution over queries* $(\boldsymbol{r}_1, \boldsymbol{r}_2, \ldots, \boldsymbol{r}_m)$ *as well as weights* $(w_1, w_2, \ldots, w_n)$ *and output*

$$T_m = \sum_{i=1}^{m} w_i \mathcal{O}(\boldsymbol{r}_i),$$

the minimum variance estimator for which $\mathbf{E}T_m = \operatorname{tr}\boldsymbol{A}$ *uniformly on* M_n *is achieved by sampling* $\{\boldsymbol{r}_i\}$ *as a collection of* m *orthogonal unit vectors and outputting*

$$T_m^\star = \frac{n}{m}\sum_{i=1}^{m} \mathcal{O}(\boldsymbol{r}_i)$$

THEOREM 3.5 (Wimmer et al. Thm 2). *Any possibly nonlinear or adaptive estimator for the trace of a matrix* $\boldsymbol{A}$ *that sequentially submits random queries* $\boldsymbol{r}_i$ *to* $\mathcal{O}$ *after seeing the previous* $i-1$ *queries needs* $\Omega(1/\epsilon)$ *queries to achieve* ϵ *mean squared error uniformly across all matrices with Frobenius norm 1.*

THEOREM 3.6 (Wimmer et al. Thm 3). *Any possibly nonlinear or adaptive estimator for the trace of a matrix* $\boldsymbol{A}$ *that sequentially submits random queries* $\boldsymbol{r}_i$ *to* $\mathcal{O}$ *after seeing the previous* $i-1$ *queries needs* $\Omega(\frac{1}{\epsilon^2}\log\frac{1}{\delta})$ *queries to output an estimator* T_m *that satisfies*

$$\mathbf{P}\big(|T - \operatorname{tr}\boldsymbol{A}| > \epsilon \operatorname{tr}\boldsymbol{A}\big) \leq \delta$$

for any rank-one positive definite matrix $\boldsymbol{A}$.

The proofs of these theorems are long and beyond the scope of this lecture. See Wimmer et al. (2014) for more details.

As we saw in the case of Schatten norm estimation, the common algorithmic setting we have isn't an oracle mapping $\boldsymbol{x} \mapsto \langle \boldsymbol{A}\boldsymbol{x}, \boldsymbol{x}\rangle$, but rather an oracle which can tell us $\boldsymbol{A}\boldsymbol{x}$ for any given $\boldsymbol{x}$. Thus, while the lower bounds of Wimmer et al. (2014) are interesting, correct, and relevant, the following open problem really gets to the heart of the fundamental problem of trace estimation as it is actually used in practice.

OPEN PROBLEM 3.7. *If we have access to an oracle* $\mathcal{O} : \boldsymbol{x} \mapsto \boldsymbol{A}\boldsymbol{x}$, *is it true that* $\Omega(\frac{1}{\epsilon^2}\log\frac{1}{\delta})$ *queries to* $\mathcal{O}$ *are needed to return a (possibly adaptive and nonlinear) estimator* T *for the trace of* $\boldsymbol{A}$ *with*

$$\mathbf{P}\big(|T - \operatorname{tr}\boldsymbol{A}| > \epsilon \operatorname{tr}\boldsymbol{A}\big) \leq \delta$$

for every positive semi-definite matrix $\boldsymbol{A}$.

References.

Dimitris Achlioptas. Database-friendly random projections. In *Proceedings of the twentieth ACM SIGMOD-SIGACT-SIGART symposium on Principles of database systems*, pages 274–281. ACM, 2001.

Ryan P Adams, Jeffrey Pennington, Matthew J Johnson, Jamie Smith, Yaniv Ovadia, Brian Patton, and James Saunderson. Estimating the spectral density of large implicit matrices. *arXiv preprint arXiv:1802.03451*, 2018.

Haim Avron. Counting triangles in large graphs using randomized matrix trace estimation. In *Workshop on Large-scale Data Mining: Theory and Applications*, volume 10, pages 10–9, 2010.

Christos Boutsidis, Petros Drineas, Prabhanjan Kambadur, Eugenia-Maria Kontopoulou, and Anastasios Zouzias. A randomized algorithm for approximating the log determinant of a symmetric positive definite matrix. *Linear Algebra and its Applications*, 533:95–117, 2017.

John D Cook. Upper and lower bounds for the normal distribution function, 2009.

Edoardo Di Napoli, Eric Polizzi, and Yousef Saad. Efficient estimation of eigenvalue counts in an interval. *Numerical Linear Algebra with Applications*, 23(4):674–692, 2016.

Gene H Golub and Charles F Van Loan. *Matrix computations*, volume 3. JHU Press, 2012.

Abdolhossein Hoorfar and Mehdi Hassani. Approximation of the lambert w function and hyperpower function. *Research report collection*, 10(2), 2007.

Farbod Roosta-Khorasani and Uri Ascher. Improved bounds on sample size for implicit matrix trace estimators. *Foundations of Computational Mathematics*, 15(5):1187–1212, 2015.

Larry Wasserman. *All of statistics: a concise course in statistical inference*. Springer Science & Business Media, 2013.

Karl Wimmer, Yi Wu, and Peng Zhang. Optimal query complexity for estimating the trace of a matrix. In *International Colloquium on Automata, Languages, and Programming*, pages 1051–1062. Springer, 2014.

David P Woodruff et al. Sketching as a tool for numerical linear algebra. *Foundations and Trends® in Theoretical Computer Science*, 10(1–2):1–157, 2014.