Randomized Linear Algebra Approaches to Estimate the Von Neumann Entropy of Density Matrices

Eugenia-Maria Kontopoulou

in collaboration with

W. Szpankowski, A, Grama & P. Drineas

Department of Computer Science
Purdue University

IEEE International Symposium on Information Theory

Vail, June 2018

Von-Neumann Entropy of Density Matrices I

Definition

Given a Density Matrix $\mathbf{R} \in \mathbb{R}^{n \times n}$, compute (exactly or approximately) the Von-Neumann entropy, $\mathcal{H}(\mathbf{R})$. A density matrix can be represented as:

$$\mathbf{R} = \sum_{i=1}^{n} p_i \psi_i \psi_i^\top,$$

where the vectors $\psi_i \in \mathbb{R}^n$ represent the pure states of a system and are pairwise orthogonal and normal, while the p_i's correspond to the probability of each state and satisfy $p_i > 0$ and $\sum_{i=1}^{n} p_i = 1$.

Letting $\Psi = [\psi_1 \ \psi_2 \ \dots \ \psi_n] \in \mathbb{R}^{n \times n}$ and $\Sigma_p = \operatorname{diag}(p_1, p_2, \dots, p_n) \in \mathbb{R}^{n \times n}$, $\mathbf{R}$ can be algebraically expressed as:

$$\mathbf{R} = \Psi \Sigma_p \Psi^\top.$$

Application: Information theory, quantum mechanics, ...

Von-Neumann Entropy of Density Matrices II

Straightforward Computation

1. Compute the probabilities of $\mathbf{R}$, $p_1, p_2, \ldots, p_n$ (e.g. computing the eigenvalue decomposition or the singular value decomposition of $\mathbf{R}$).
2. Compute the Von-Neumann Entropy of $\mathbf{R}$:

$$\mathcal{H}(\mathbf{R}) = -\sum_{i=1}^{n} p_i \log p_i.$$

Time Complexity: $\mathcal{O}(n^3)$.

Prohibitive for Large Data!!!!

Mathematical Manipulation of $\mathcal{H}(\mathbf{R})$

Consider the function $h(x) = x \log x \in \mathbb{R}$.

$$
\begin{aligned}
h(\mathbf{R}) &= \mathbf{R} \log \mathbf{R} \\
&= \Psi \Sigma_p \Psi^\top \log(\Psi \Sigma_p \Psi^\top) \\
&= \Psi \Sigma_p \log(\Sigma_p) \Psi^\top \\
&= \Psi h(\Sigma_p) \Psi^\top
\end{aligned}
$$

$$
\begin{aligned}
\mathcal{H}(\mathbf{R}) &= -\sum_i p_i \log p_i \\
&= -\mathbf{Tr}\left(h(\Sigma_p)\right) \\
&= -\mathbf{Tr}\left(\Psi^\top \Psi h(\Sigma_p)\right) \\
&= -\mathbf{Tr}\left(\Psi h(\Sigma_p) \Psi^\top\right) \\
&= -\mathbf{Tr}\left(h(\mathbf{R})\right)
\end{aligned}
$$

Two Approaches

1. Using a Taylor expansion for the logarithm we can further manipulate $\mathcal{H}(\mathbf{R})$.
2. Approximate $h(\mathbf{R})$ with Chebyshev Polynomials.

Two Randomized Numerical Linear Algebra tools

1. Power method with provable bounds.
2. Randomized trace estimators.

Lemma

Let $\mathbf{R} \in \mathbb{R}^{n \times n}$ *be a density matrix whose probabilities lie in the interval* $[\ell, u]$*, for some* $0 < \ell \leq u \leq 1$*. Then,*

$$\mathcal{H}(\mathbf{R}) = \log u^{-1} + \underbrace{\sum_{k=1}^{\infty} \frac{\mathbf{Tr}\left(\mathbf{R}(\mathbf{I} - u^{-1}\mathbf{R})^k\right)}{k}}_{\Delta} .$$

We estimate u using the power method[1], $\mathbf{Tr}\left(\mathbf{R}(\mathbf{I} - u^{-1}\mathbf{R})^k\right)$ using a Gaussian trace estimator[2] and we truncate Δ by keeping the first m terms.

---

[1] L. Trevisan (2011), *``Graph Partitioning and Expanders''*

[2] H. Avron and S. Toledo (2011), *``Randomized Algorithms for Estimating the Trace of an Implicit Symmetric Positive Semi-definite Matrix''*

Algorithm 1

Input: $R \in \mathbb{R}^{n \times n}$, accuracy parameter $\epsilon > 0$, integer $m > 0$.

Output: $\widehat{\mathcal{H}(R)}$, the approximation to $\mathcal{H}(R)$.

1. Approximate u, an estimate for the largest probability, via the power method.
2. Create $s = \lceil 20 \log(2/\delta)/\epsilon^2 \rceil$ i.i.d random Gaussian vectors, $g_1, g_2, \ldots, g_s$.
3. Compute $\widehat{\mathcal{H}(R)}$ as:

$$\widehat{\mathcal{H}(R)} = \log u^{-1} + \sum_{k=1}^{m} \frac{\frac{1}{s}\sum_{i=1}^{s}\left(g_i^\top \; R \; \left(\mathbf{I} - u^{-1}\mathbf{R}\right)^k \; g_i\right)}{k}.$$

Theorem

Let $\mathbf{R}$ *be a density matrix such that all probabilities* p_i*,* $i = 1 \ldots n$ *satisfy* $0 < \ell \leq p_i$*. Let* u *be computed as in Algorithm 1 and let* $\widehat{\mathcal{H}(\mathbf{R})}$ *be the output of Algorithm 1 on inputs* $\mathbf{R}$*,* m*, and* $\epsilon < 1$*; Then, with probability at least* $1 - 2\delta$*,*

$$\left|\widehat{\mathcal{H}(\mathbf{R})} - \mathcal{H}\left(\mathbf{R}\right)\right| \leq 2\epsilon\mathcal{H}\left(\mathbf{R}\right),$$

by setting $m = \lceil \frac{u}{\ell} \log\left(1/\epsilon\right) \rceil$*.*

Running Time

$$\mathcal{O}\left(\left(\frac{u}{\ell} \cdot \frac{\log\left(1/\epsilon\right)}{\epsilon^2} + \log n\right) \cdot \log\left(1/\delta\right) \cdot nnz(\mathbf{R})\right).$$

Lemma

We can approximate $h(x) = x \log x$ in the interval $(0, u]$ by

$$f_m(x) = \sum_{w=0}^{m} \alpha_w \mathcal{T}_w(x),$$

where $\mathcal{T}_w(x) = \cos(w \cdot \arccos((2/u)x - 1))$, the Chebyshev polynomials of the first kind for $w > 0$ and,

$$\alpha_0 = \frac{u}{2}\left(\log\frac{u}{4} + 1\right), \quad \alpha_1 = \frac{u}{4}\left(2\log\frac{u}{4} + 3\right), \quad \text{and} \quad \alpha_w = \frac{(-1)^w u}{w^3 - w} \text{ for } w \geq 2.$$

For any $m \geq 1$,

$$|h(x) - f_m(x)| \leq \frac{u}{2m(m+1)} \leq \frac{u}{2m^2},$$

for $x \in [0, u]$.

Using the Lemma we approximate $\mathcal{H}(\mathbf{R})$ by $\widehat{\mathcal{H}(\mathbf{R})}$ as follows:

$$\begin{aligned} \mathcal{H}(\mathbf{R}) &= -\mathbf{Tr}\left(h(\mathbf{R})\right) \\ &\approx -\mathbf{Tr}\left(f_m(\mathbf{R})\right) \\ &\approx -\frac{1}{s}\sum_{i=1}^{s} \mathbf{g}_i^\top f_m(\mathbf{R})\mathbf{g}_i \\ &= \widehat{\mathcal{H}(\mathbf{R})} \end{aligned}$$

We estimate u using the power method and $\mathbf{Tr}\left(f_m(\mathbf{R})\right)$ using a Gaussian trace estimator. We compute the scalars $\mathbf{g}_i^\top f_m(\mathbf{R})\mathbf{g}_i$ using the Clenshaw algorithm.

Algorithm 2

Input: $R \in \mathbb{R}^{n\times n}$, accuracy parameter $\epsilon > 0$, integer $m > 0$.

Output: $\widehat{\mathcal{H}(R)}$, the approximation to $\mathcal{H}(R)$.

1. Approximate u, an estimate for the largest probability, via the power method.
2. Create $s = \lceil 20\log(2/\delta)/\epsilon^2 \rceil$ i.i.d random Gaussian vectors, $g_1, g_2, \ldots, g_s$.
3. Compute $\widehat{\mathcal{H}(R)}$ as:

$$\widehat{\mathcal{H}(R)} = -\frac{1}{s}\sum_{i=1}^{s} g_i^\top f_m(R) g_i.$$

Theorem

Let $\mathbf{R}$ *be a density matrix such that all probabilities* p_i, $i = 1 \ldots n$ *satisfy* $0 < \ell \leq p_i$. *Let* u *be computed as in Algorithm 2 and let* $\widehat{\mathcal{H}(\mathbf{R})}$ *be the output of Algorithm 2 on inputs* $\mathbf{R}$, m, *and* $\epsilon < 1$*; Then, with probability at least* $1 - 2\delta$,

$$\left|\widehat{\mathcal{H}(\mathbf{R})} - \mathcal{H}\left(\mathbf{R}\right)\right| \leq 3\epsilon\mathcal{H}\left(\mathbf{R}\right),$$

by setting $m = \sqrt{\frac{u}{2\epsilon\ell\ln(1/(1-\ell))}}$.

Running Time

$$\mathcal{O}\left(\left(\sqrt{\frac{u}{\ell\ln(1/(1-\ell))}}\cdot\frac{1}{\epsilon^{2.5}} + \ln(n)\right)\cdot\ln(1/\delta)\cdot nnz(\mathbf{R})\right).$$

Assume that the density matrix $\mathbf{R} \in \mathbb{R}^{n \times n}$, has at most k non-zero probabilities, p_i. This means that at most k of its states are pure.

Issue & Solution

- x $n - k$ probabilities are zero $\rightarrow$ Chebyshev/Taylor approaches are not working.
- ✓ Project to a smaller full-dimensional space $\rightarrow$ Random Projections (Gaussian, sub-sampled randomized Hadamard transform, input sparsity transform, Hartley transform).

Algorithmic Scheme

1. Construct the random projection matrix $\Pi \in \mathbb{R}^{n \times s}$.
2. Compute $\tilde{\mathbf{R}} = \mathbf{R}\Pi \in \mathbb{R}^{n \times s}$.
3. Compute at most k probabilities, $\tilde{p}_i, \ i = 1, \ldots, k$, of $\tilde{\mathbf{R}}$.
4. Compute $\widehat{\mathcal{H}(\tilde{\mathbf{R}})}$, the estimate to $\widehat{\mathcal{H}(\mathbf{R})}$.

Theorem

Let $\mathbf{R}$ *be a density matrix with at most* $k \ll n$ *non-zero probabilities and let* $\epsilon < 1/2$ *be an accuracy parameter. Then, with probability at least 0.9, the output of the algorithmic scheme satisfies*

$$\left|p_i^2 - \tilde{p}_i^2\right| \leq \epsilon p_i^2$$

for all $i = 1 \dots k$*. Additionally,*

$$\left|\mathcal{H}(\mathbf{R}) - \widehat{\mathcal{H}(\mathbf{R})}\right| \leq \sqrt{\epsilon}\mathcal{H}(\mathbf{R}) + \sqrt{\frac{3}{2}}\epsilon.$$

Running Time

Using the input sparsity transform [3] random projection method:

$$\mathcal{O}\left(nnz(\mathbf{R}) + \frac{nk^4}{\epsilon^4}\right).$$

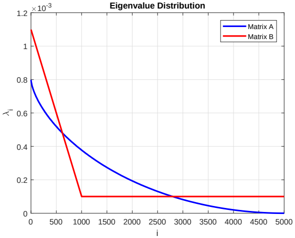
[3] K. Clarkson, D. Woodruff (2013), *``Low Rank Approximation and Regression in Input Sparsity Time''*
E. Kontopoulou, G. Dexter, W. Szpankowski, A. Grama & P. Drineas (2018), "*Randomized Linear Algebra Approaches to Estimate the Von Neumann Entropy of Density Matrices*"

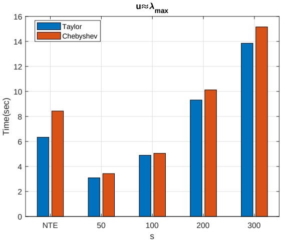
Experiment 1 - I

Running Time

Random density matrices of size $5,000 \times 5,000$

- ✓ **Matrix A:** exponentially decaying probabilities.
- ✓ **Matrix B:** $1,000$ linearly decaying probabilities.





Parameters

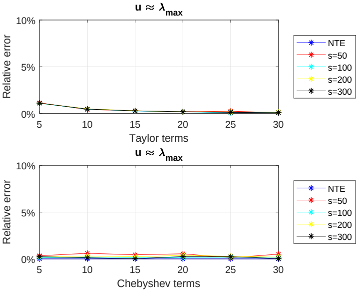
- ✓ Polynomial terms: $m = [5 : 5 : 30]$
- ✓ Gaussian vectors: $s = \{50, 100, 200, 300\}$
- ✓ Largest probability: $u \approx \lambda_{max}$

Notes

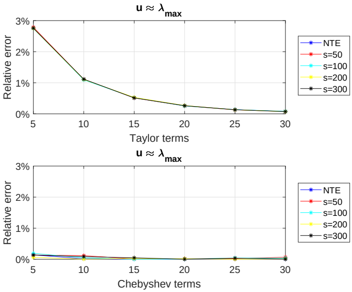
- Exact computation: 1.5 minutes.
- Approximation of λ_{max}: < 1 second.

Parameters

- ✓ Polynomial terms: $m = [5 : 5 : 30]$
- ✓ Gaussian vectors: $s = \{50, 100, 200, 300\}$
- ✓ Largest probability: $u \approx \lambda_{max}$



Matrix A



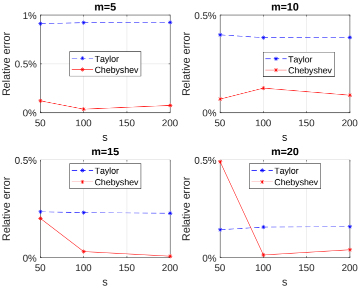
Matrix B

Experiment 2

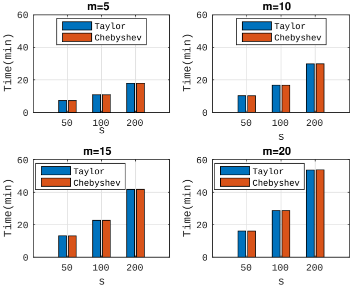
Random density matrix of size $30,000 \times 30,000$

✓ Polynomial terms: $m = [5 : 5 : 20]$

✓ Largest probability: $u \approx \lambda_{max}$



Gaussian vectors: $s = [50 : 50 : 200]$



Gaussian vectors: $s = \{50, 100, 200\}$

Notes

- Exact computation: 5.6 hours.
- Approximation of λ_{max}: 3.6 minutes.

Thank you!

Questions?

Bibliography

- H. Avron. S. Toledo, *``Randomized algorithms for estimating the trace of an implicit symmetric positive semi-definite matrix´´*, in Journal of the ACM, 58(2):8, 2011
- K. L. Clarkson, D. P. Woodruff, *``Low Rank Approximation and Regression in Input Sparsity Time´´*, in Proceedings of the 45th annual ACM Symposium on Theory of Computing, p. 81-90, ACM Press, 2013.
- I. Han, D. Malioutov, J. Shin, *``Large-scale Log-determinant Computation through Stochastic Chebyshev Expansions´´*, in Proceedings of the 32nd International Conference on Machine Learning, 37:908-917, 2015.
- E. Kontopoulou, G. Dexter, W. Szpankowski, A. Grama, P. Drineas, *``Randomized Linear Algebra Approaches to Estimate the Von Neumann Entropy of Density Matrices´´*, ArXiv, 2018
- F. Roosta-Khorasani, U. Ascher, *``Improved bounds on sample size for implicit matrix trace estimators´´*, in Foundations of Computational Mathematics, 15:1187-1212, 2015