Randomized algorithms to approximate logarithm-based matrix functions

Eugenia-Maria Kontopoulou & Petros Drineas

Computer Science, Purdue University

LogDet Problem

Given a Symmetric Positive Definite matrix $\mathbf{A} \in \mathbb{R}^{n\times n}$, compute (exactly or approximately) the $\log\det(\mathbf{A})$.

Application: Maximum likelihood estimations, Gaussian processes prediction, $\log\det$-divergence metric, barrier functions in interior point methods . . .

Von-Neumann Entropy

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 is represented by the statistical mixture of pure states and has the form

$$\mathbf{R} = \sum_{i=1}^{n} p_i \psi_i \psi_i^\top = \Psi \Sigma_p \Psi^\top \in \mathbb{R}^{n\times n},$$

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

Application: Information theory, quantum mechanics, . . .

Approximation via Taylor

Let $\mathbf{A} \in \mathbb{R}^{n\times n}$ be an SPD matrix whose eigenvalues lie in the interval $(\theta_1, 1)$, for some $0 < \theta_1 < 1$. Let $\mathbf{C} = \mathbf{I}_n - \mathbf{A}$. Then (using the Taylor expansion of $\log \mathbf{C}$),

$$\log\det(\mathbf{A}) \approx -\sum_{k=1}^{m} \frac{\mathbf{Tr}\left(\mathbf{C}^k\right)}{k}.$$

Let $\mathbf{R} \in \mathbb{R}^{n\times n}$ be an SPD matrix with unit trace, whose eigenvalues lie in the interval $[\ell, u]$, for some $0 < \ell \le u \le 1$. Let $\mathbf{C} = \mathbf{I} - u^{-1}\mathbf{R}$ Then (using the Taylor expansion of $\log \mathbf{C}$),

$$\mathcal{H}(\mathbf{R}) \approx \log u^{-1} + \sum_{k=1}^{m} \frac{\mathbf{Tr}\left(\mathbf{R}\mathbf{C}^k\right)}{k}.$$

Approximation via Chebyshev

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

$$f_m(x) = \sum_{t=0}^{m} \alpha_t \mathcal{T}_t(x),$$

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

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

Then

$$\mathcal{H}(\mathbf{R}) \approx -\mathbf{Tr}\left(f_m(\mathbf{R})\right)$$

Gaussian trace estimator

We use Gaussian trace estimators to estimate the trace of powers of $\mathbf{C}$. An (ϵ, δ) Gaussian trace estimator for an SPD $\mathbf{A} \in \mathbb{R}^{n\times n}$ is the matrix:

$$\mathbf{G} = \frac{1}{p}\sum_{i=1}^{p} g_i^\top \mathbf{A}\, g_i,$$

where the g_i's are p independent random vectors whose entries are i.i.d. standard normal variables. [AvronToledo2011]

Clenshaw's Algorithm

We use Clenshaw's algorithm to evaluate Chebyshev polynomials with matrix inputs. Clenshaw's algorithm is a recursive approach with base cases $b_{m+2}(x) = b_{m+1}(x) = 0$ and the recursive step (for $k = m, m-1, \ldots, 0$) which in our case is: $b_k(x) = \alpha_k + 2\left(\frac{2}{u}x - 1\right) b_{k+1}(x) - b_{k+2}(x)$. Then:

$$f_m(x) = \frac{1}{2}\left(\alpha_0 + b_0(x) - b_2(x)\right).$$

Power Method

We use the Power Method algorithm to estimate the largest eigenvalue of the matrix. We prove the following, building upon [Trevisan2011]:

Let $\tilde{p}_1$ be the output of Power Method with $q = \lceil 4.82\log(1/\delta)\rceil$ restarts of the algorithm and $t = \lceil \log\sqrt{4n} \rceil$ iterations before each restart. Then, with probability at least $1 - \delta$,

$$\frac{1}{6}p_1 \le \tilde{p}_1 \le p_1.$$

Algorithms

Algorithm 1 - LogDet via Taylor Approximation

Input: $\mathbf{A} \in \mathbb{R}^{n\times n}$ with eigenvalues lie in $(\theta_1, 1)$ where $\theta_1 > 0$, accuracy parameter $\epsilon > 0$, integer $m > 0$.

Output: $\widehat{\log\det}(\mathbf{A})$, the approximation to the $\log\det(\mathbf{A})$.

1. $\mathbf{C} = \mathbf{I}_n - \mathbf{A}$
2. Create $p = \lceil 20\log(2/\delta)/\epsilon^2 \rceil$ i.i.d random Gaussian vectors, $g_1, g_2, \ldots, g_p$.
3. Estimate $\widehat{\log\det}$ as:

$$\widehat{\log\det}(\mathbf{A}) = \sum_{k=1}^{m}\left(\frac{1}{p}\sum_{i=1}^{p} g_i^\top \mathbf{C}^k g_i\right).$$

Algorithm 2 - Entropy via Taylor Approximation

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

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

1. Compute $\hat{p}_1$, the estimation of the largest singular value of R, using power method.
2. Set $u = \min\{1, 6\hat{p}_1\}$
3. $C = I_n - u^{-1}R$
4. Create $p = \lceil 20\log(2/\delta)/\epsilon^2 \rceil$ i.i.d random Gaussian vectors, $g_1, g_2, \ldots, g_p$.
5. Compute $\widehat{\mathcal{H}_T(R)}$ as:

$$\widehat{\mathcal{H}_T(R)} = \log u^{-1} + \frac{1}{p}\sum_{i=1}^{p}\sum_{k=1}^{m} \frac{g_i^\top\, R\, C^k\, g_i}{k}.$$

Algorithm 3 - Entropy via Chebyshev Approximation

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

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

1. Compute $\hat{p}_1$, the estimation of the largest singular value of R, using power method.
2. Set $u = \min\{1, 6\hat{p}_1\}$
3. Create $p = \lceil 20\log(2/\delta)/\epsilon^2 \rceil$ i.i.d random Gaussian vectors, $g_1, g_2, \ldots, g_p$.
4. Compute $\widehat{\mathcal{H}_C(R)}$ as:

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

Theoretical Results

We **prove** that:

✓ $\widehat{\log\det}(\mathbf{A})$ is an (ϵ, δ)-estimator of $\log\det(\mathbf{A})$ and can be computed in

$$\mathcal{O}\left(\frac{\log(1/\epsilon)\log(1/\delta)}{\epsilon^2 \cdot \theta_1} \cdot nnz(\mathbf{A})\right)$$

✓ $\widehat{\mathcal{H}_T(\mathbf{R})}$ is an (ϵ, δ)-estimator of $\mathcal{H}(\mathbf{R})$ and can be computed in

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

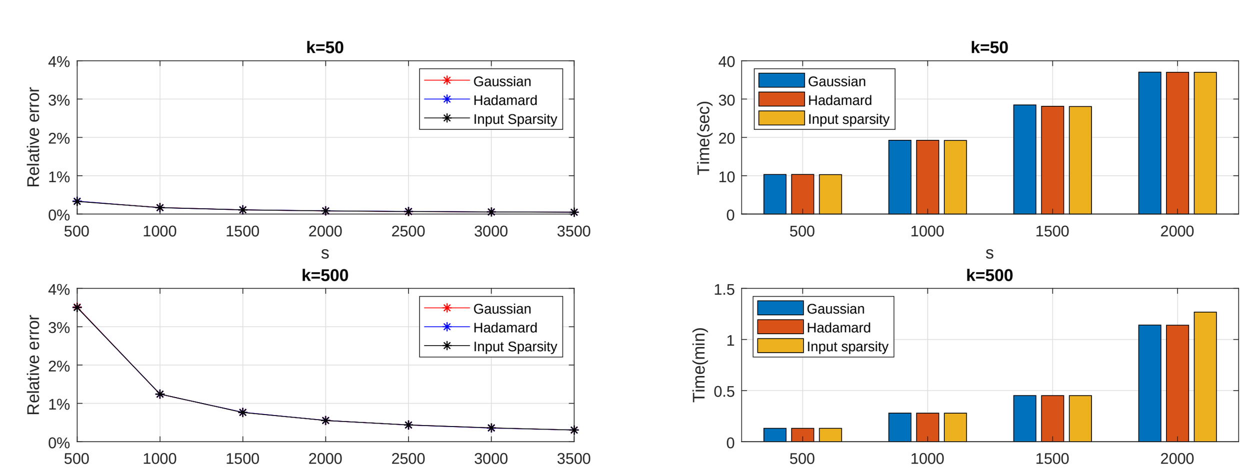
✓ $\widehat{\mathcal{H}_C(\mathbf{R})}$ is an (ϵ, δ)-estimator of $\mathcal{H}(\mathbf{R})$ and can be computed in

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

Future Directions

- **Eigenvalue distribution:** How much is the relative error affected by the distribution of the eigenvalues when we use the polynomial-based algorithms?
- **Low rank Density Matrices:** Polynomial-based algorithms do not work. We can use **Random Projections** to approximate the Von-Neumann Entropy of low rank Density Matrix. Theoretically we get additive (ϵ, δ)-estimators. **Considerations:** Fast construction of the random projection matrix, meaningful bounds...

Matrix of size $16,384 \times 16,384$



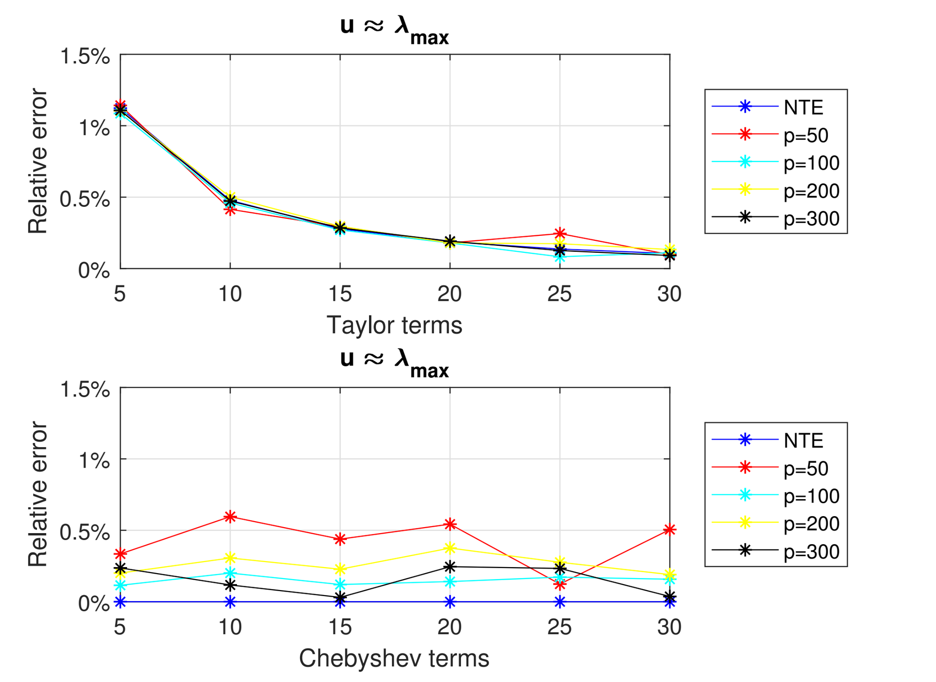
$s = [500 : 500 : 3500]$. $s = [500 : 500 : 2000]$

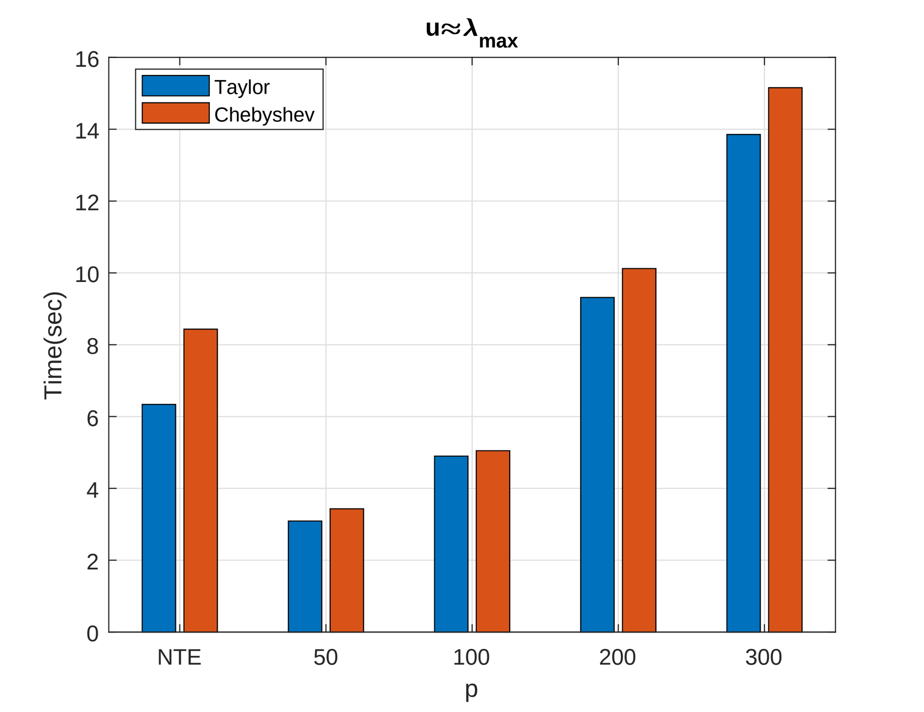
Exact computation: 1.6 minutes (rank 50), 20 minutes (rank 500).

Experiments - Entropy Approximation

All experiments ran at Purdue's Snyder cluster using 1 node.

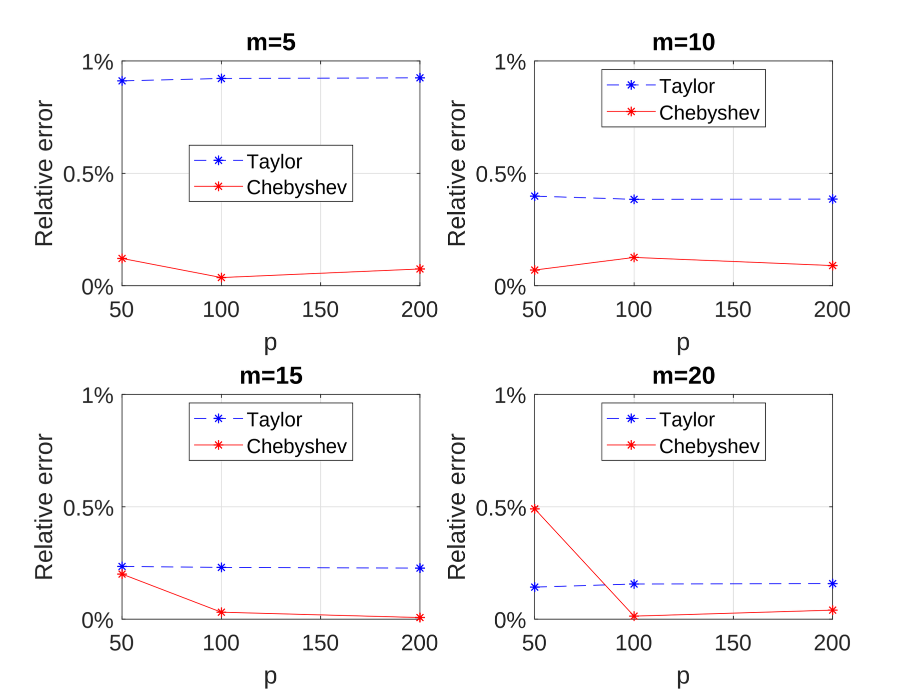
Matrix of size $5,000 \times 5,000$:

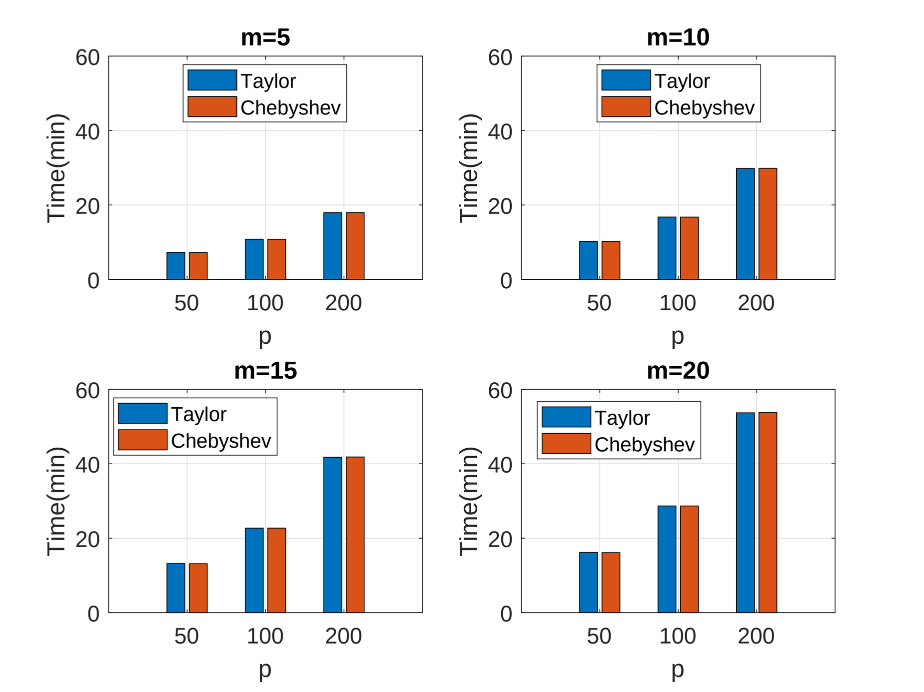




Exact computation: 1.5 min

Matrix of size $30,000 \times 30,000$, **and** $p = [50 : 50 : 200]$:





Exact computation: 5.6 hours

Citations

- C. Boutsidis, P. Drineas, P. Kambadur, E. Kontopoulou, A. Zouzias, ***"A Randomized Algorithm for Approximating the Log Determinant of a Symmetric Positive Definite Matrix"***, in Linear Algebra and its Applications, 533:95-117, 2017
- E. Kontopoulou, A. Grama, W. Szpankowski and P. Drineas, ***Randomized Linear Algebra Approaches to Estimate the Von-Neumann Entropy of Density Matrices***, available in ArXiv
- H. Avron, S. Toledo. ***Randomized Algorithms for Estimating the Trace of an Implicit Symmetric Positive Semi-definite Matrix***. J. ACM, 58(2):8,2011
- L. Trevisan, ***Graph Partitioning and Expanders***, Handout 7, 2011