Accelerating Matrix Trace Estimators by Aitken's Δ^2 Process

Vassilis Kalantzis, Georgios Kollias, Shashanka Ubaru, and Theodoros Salonidis

May 2023

EPrint ID: 2023.1

IBM Research
Thomas J. Watson Research Center

Preprints available from:
https://researcher.watson.ibm.com/researcher/view.php?person=ibm-vkal

ACCELERATING MATRIX TRACE ESTIMATION BY AITKEN'S Δ^2 PROCESS

Vassilis Kalantzis *Georgios Kollias* *Shashanka Ubaru* *Theodoros Salonidis*

IBM Research, Thomas J. Watson Research Center, Yorktown Heights, NY, USA

ABSTRACT

We present an algorithm to estimate the trace of symmetric matrices that are available only via Matrix-Vector multiplication. The proposed algorithm constructs a series of trace estimates by applying the probing technique with an increasing number of vectors. These estimates are then treated as a converging sequence whose limit is the sought matrix trace, and we apply Aitken's Δ^2 process to accelerate its convergence to the trace limit. Numerical experiments performed on covariance matrices demonstrate the competitiveness of the proposed scheme versus probing and randomized trace estimators.

1. INTRODUCTION

Computing the sum of the diagonal elements (trace) of a $n \times n$ symmetric matrix $\mathbf{A}$ is an important computational task that arises in a wide range of scientific applications, including uncertainty quantification, lattice quantum chromodynamics, network theory, and protein folding, e.g., see [1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11]. In addition, several machine learning applications require the computation of the trace of covariance matrices. For example, the Fréchet Inception Distance (FID), a performance metric to evaluate the quality of Generalised Adversarial Networks, requires the computation of the trace of sample covariance matrices associated with distributions of real and engineered datasets [12, 13]. Similarly, the trace estimation of precision matrices appears when fitting the Matérn covariance model to a Gaussian process via maximum likelihood estimation [14]. Finally, computing the trace of covariance matrices allows the quantification of the total variance of the underlying data-collection [1].

In this paper we focus on scenarios where $\mathbf{A}$ is implicitly-defined i.e., $\mathbf{A}$ is not formed explicitly but we can compute Matrix-Vector (MV) products of the form $\boldsymbol{y} = \mathbf{A}\boldsymbol{z},\ \boldsymbol{z} \in \mathbb{R}^n$. For example, let $\mathbf{A} = \dfrac{1}{n}\mathbf{X}\mathbf{X}^T$ denote the sample covariance matrix associated with the matrix $\mathbf{X}$ containing n observed data of dimension p. Forming the matrix $\mathbf{A}$ explicitly requires the storage of n^2 scalars and $2pn^2$ floating-point operations, which becomes impractical for large data-collections. On the other hand, we can compute MV products with $\mathbf{A}$ through computing MV products with the matrices $\mathbf{X}$ and $\mathbf{X}^T$.

The simplest approach to compute the trace $\mathtt{tr}(\mathbf{A})$ of the matrix $\mathbf{A}$ is to compute each individual diagonal entry individually, i.e., $\mathtt{tr}(\mathbf{A}) = \sum_{j=1}^{j=n} \boldsymbol{e}_j^T \mathbf{A} \boldsymbol{e}_j$, where $\boldsymbol{e}_j \in \mathbb{R}^n$ denotes the jth column of the $n \times n$ identity matrix. This approach requires n MV products with matrix $\mathbf{A}$ and becomes impractical for anything but very small matrix size n.

Instead, practical algorithms settle for an approximation of the form $\mathtt{tr}(\mathbf{A}) \approx \sigma \sum_{j=1}^{j=k} \boldsymbol{z}_j^T \mathbf{A} \boldsymbol{z}_j,\ \sigma > 0$, where the entries of the n-dimensional vectors $\{\boldsymbol{z}_j\}_{j=1}^{j=k}$ are chosen either randomly or deterministically. For example, when each entry of $\boldsymbol{z}_j$ is ±1 with equal probability (i.e., Rademacher vector) and $\sigma = 1/k$, the above Monte Carlo (MC) trace estimator is a minimum variance[1] unbiased estimator of $\mathtt{tr}(\mathbf{A})$ [15], and converges as $O\left(1/\sqrt{k}\right)$. Standard MC approaches do not take under consideration any special properties of matrix $\mathbf{A}$, e.g., decay in the magnitude of the off-diagonal entries. An alternative to random vectors is to set the vectors $\{\boldsymbol{z}_j\}_{j=1}^{j=k}$ as columns of the Hadamard matrix of the appropriate size or as linear combinations of the $n \times n$ identity matrix. This choice of vectors has the property that the error in the approximation of $\mathtt{tr}(\mathbf{A})$ only comes from certain non-zero entries, e.g., those located on diagonals which are multiple of k. We will refer to such procedures as *probing* [16, 17]. A well-known limitation of probing versus MC estimators is the discarding of all previous computational efforts every time a new approximation of $\mathtt{tr}(\mathbf{A})$ is computed. This has been studied in [4, 9] where it was suggested to use hierarchical probing.

The algorithm proposed in this paper alleviates the incremental nature of probing by applying a series transformation process. In particular, under the assumption that the estimates $\chi_0, \chi_1, \ldots$, produced by the probing trace estimator converge (approximately) linearly to $\mathtt{tr}(\mathbf{A})$, we can treat the former as part of a convergent sequence which we accelerate by transforming it to another sequence which has the same limit (i.e., $\mathtt{tr}(\mathbf{A})$), but converges to it at a super-linear rate. This is a series acceleration problem, and the algorithm discussed in this paper accelerates the sequence $\chi_0, \chi_1, \ldots$, by applying Aitken's delta-squared (Δ^2) process [18, 19]. Aitken's extrapolation method has been considered previously in the context of estimating the trace of the matrix inverse by extrapolating the prediction of the moment $\boldsymbol{x}^T \mathbf{A}^{-1} \boldsymbol{x}$ [20]; see also [21]. Nonetheless, the latter work is quite different than the algorithm proposed in this paper as we perform no moment extrapolation; rather we apply Aitken's Δ^2 process to a sequence of trace estimates obtained by a probing trace estimator.

Without loss of generality, throughout the rest of this paper we will assume that the size of matrix $\mathbf{A}$ satisfies $n = 2^j$ for some $j \in \mathbb{N}$. Lowercase bold letters, $\boldsymbol{x}$, will denote vectors, while uppercase bold letters, $\mathbf{X}$, will represent matrices.

Emails: vkal@ibm.com, gkollias@us.ibm.com, shashanka.ubaru@ibm.com, tsaloni@us.ibm.com

[1] Over the field of real random vectors.

Section 2 describes the basic procedure following by the probing technique. Section 3 improves the accuracy of the probing trace estimator by Aitken's extrapolation method. Section 4 illustrates the proposed method on a model problem from uncertainty quantification. Finally, Section 5 presents our concluding remarks.

2. APPROXIMATING $\mathtt{TR}(A)$ BY PROBING

Definition 1. *For any integer* $0 \leq i \leq \mathtt{log}_2(n)$*, define the integer variable*

$$k_i = 2^i,$$

where $\mathtt{log}_2$ *denotes the binary logarithm. We then define the approximation* χ_i *of* $\mathtt{tr}(\mathbf{A})$*:*

$$\chi_i = \sum_{j=1}^{j=k_i} \boldsymbol{z}_j^T \mathbf{A} \boldsymbol{z}_j,$$

where all entries of $\boldsymbol{z}_j \in \mathbb{R}^n$ *are equal to zero except for those indexed by* $(i-1)k_i + j,\ i = 1, \ldots, \dfrac{n}{k_i}$*, which are equal to one.*

We will denote by $\mathbf{Z}_{k_i} = [\boldsymbol{z}_1, \ldots, \boldsymbol{z}_{k_i}]$ the matrix whose jth column is equal to $\boldsymbol{z}_j,\ j = 1, \ldots, k_i$. For example, let $n = 4$. The matrices $\mathbf{Z}_1,\ \mathbf{Z}_2$, and $\mathbf{Z}_4$, are equal to

$$\mathbf{Z}_1 = \begin{pmatrix} 1 \\ 1 \\ 1 \\ 1 \end{pmatrix},\ \mathbf{Z}_2 = \begin{pmatrix} 1 & 0 \\ 0 & 1 \\ 1 & 0 \\ 0 & 1 \end{pmatrix},\ \text{and}\ \mathbf{Z}_4 = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix},$$

respectively.

Let now the sum of the entries of the ηth super-diagonal of $\mathbf{A}$ be denoted by $\varepsilon_\eta = \sum_{p=1}^{p=n-\eta} \mathbf{A}_{p,\eta+p}$. The estimation χ_i of $\mathtt{tr}(\mathbf{A})$ can be written as:

$$\begin{aligned} \chi_i &= \mathtt{tr}(\mathbf{Z}_{k_i}^T \mathbf{A} \mathbf{Z}_{k_i}) \\ &= \mathtt{tr}(\mathbf{A}) + 2\left(\varepsilon_{k_i} + \varepsilon_{2k_i} + \ldots\right) \\ &= \mathtt{tr}(\mathbf{A}) + \sum_{j=1}^{j=(n/k_i)-1} 2\varepsilon_{jk_i}. \end{aligned} \tag{1}$$

Therefore, the error in the approximation of the $\mathtt{tr}(\mathbf{A})$ by the estimate $\chi_i = \sum_{j=1}^{j=k_i} \boldsymbol{z}_j^T \mathbf{A} \boldsymbol{z}_j$ stems only from those hyper-diagonals whose index is a multiple of the integer 2^i. Moreover, the hyper-diagonals which contribute to the approximation error of the estimate χ_i are a super-set of those hyper-diagonals which contribute to the approximation error of the estimate χ_{i+1}.

Proposition 1. *Let the entries of matrix* $\mathbf{A}$ *be finite and assume that its off-diagonal entries have the same sign. Then, the sequence* $\chi_0,\ \chi_1,\ \ldots$*, converges monotonically to* $\mathtt{tr}(\mathbf{A})$.

Proof. Since $\mathbf{A}$ has finite entries, the sequence $\chi_0,\ \chi_1,\ \ldots$, is bounded. Following (1), the hyper-diagonals which contribute to the approximation error of the estimate χ_i are a super-set of the hyper-diagonals which contribute to the approximation error of the estimate χ_{i+1}. Since $\mathbf{Z}_{k_i}$ has only non-negative entries and the off-diagonal entries of $\mathbf{A}$ have the same sign, it follows that the sequence $\chi_0,\ \chi_1,\ \ldots$, is either non-increasing or non-decreasing, and converges to $\mathtt{tr}(\mathbf{A})$. □

Matrices whose off-diagonal entries satisfy the above properties can be found in our numerical experiments as well as [5, 10, 22] and references therein.

The probing procedure is summarized in Algorithm 1. The procedure terminates when the relative difference between the two most recent trace approximations is below an acceptable threshold $\epsilon \in \mathbb{R}$. Notice that Algorithm 1 terminates after at most $\mathtt{log}_2(n)$ iterations, i.e., $\chi_{\mathtt{log}_2(n)}$ is the exact trace since $\mathbf{Z}_n$ is equal to the $n \times n$ identity matrix. The probing estimate χ_i of $\mathtt{tr}(\mathbf{A})$ requires the summation of k_i scalar (dot) products between $\boldsymbol{z}_j^T$ and $\boldsymbol{y}_j = \mathbf{A}\boldsymbol{z}_j,\ j = 1, \ldots, k_i$. Assuming that the probing trace estimator converges with χ_i as the latest estimation of $\mathtt{tr}(\mathbf{A})$, Algorithm 1 requires $2^0 + 2^1 + \ldots + 2^i = 2^{i+1} - 1$ MV products of the form $\boldsymbol{y} = \mathbf{A}\boldsymbol{z},\ \boldsymbol{z} \in \mathbb{R}^n$.

Algorithm 1 Probing trace estimator

```
1. Compute $\chi_0 = \mathbf{1}^T \mathbf{A} \mathbf{1}$, where $\mathbf{1} \in \mathbb{R}^n$ is the vector of all ones. Set $i = 0$ and a tolerance $\epsilon \in (0, 1)$.
do
    2. Set $i = i + 1,\ k_i = 2^i$
    3. Compute $\chi_i = \mathtt{tr}(\mathbf{Z}_{k_i}^T \mathbf{A} \mathbf{Z}_{k_i})$
while $|\chi_i - \chi_{i-1}| > \epsilon\chi_i$ and $i \leq \mathtt{log}_2(n)$
```

Following the above discussion, it becomes evident that computing the probing estimate χ_{i+1} requires as much work as computing all[2] previous i probing terms together. Therefore, the computation of each progressive term becomes increasingly more expensive. As a remedy, in the next section we study the use of sequence acceleration techniques in order to reduce the computational cost of Algorithm 1.

3. ACCELERATING PROBING TRACE ESTIMATORS BY AITKEN'S Δ^2 PROCESS

In this section we consider the application of a sequence acceleration technique in order to accelerate the convergence rate of probing matrix trace estimators. The idea behind sequence acceleration techniques is to transform a slowly convergent sequence[3] $\{t_n\}_{n\in\mathbb{N}}$ into a new sequence $\{\widehat{t}_n\}_{n\in\mathbb{N}}$ which converges to the same limit faster than the original sequence [23].

The literature on sequence acceleration techniques spans several centuries but research on the topic was revived during the initial era of digital computers due to Wynn's Epsilon Method [24]. A survey of modern sequence acceleration techniques can be found in [25]. Richardson's extrapolation method

[2] A more efficient implementation is possible if partial MV products are stored in memory.

[3] Note that $\{t_n\}_{n\in\mathbb{N}}$ can also be non-scalars, e.g., vectors, matrices, or tensors.

[19] and Aitken's Δ^2 process [18] are among the most well-known extrapolation-based methods used for accelerating the rate of linearly converging sequences.

Definition 2. *A convergent sequence $\{t_i\}_{i=0}^{\infty}$ converges linearly, with rate $\mu \in (0,1)$, when*

$$\lim_{i\to\infty} \frac{|t_{i+1}-t^*|}{|t_i-t^*|} = \mu.$$

Unlike Richardson extrapolation, Aitken's Δ^2 process can be applied even when the rate of convergence is unknown, since the latter does not appear in the update of $\widehat{t}_i$. For this reason, in this paper we focus exclusively on Aitken's Δ^2 process.

Definition 3. *Given a sequence $\{t_i\}_{i=0}^{\infty}$, Aitken's Δ^2 process generates a new sequence $\{\widehat{t}_i\}_{i=0}^{\infty}$ such that*

$$\begin{aligned}\widehat{t}_i &= t_i - \frac{(t_{i+1}-t_i)^2}{t_{i+2}-2t_{i+1}+t_i}\\ &= t_i - \frac{(t_{i+1}-t_i)^2}{(t_{i+2}-t_{i+1})-(t_{i+1}-t_i)}.\end{aligned}$$

Under the assumption that $\{t_i\}_{i=0}^{\infty}$ converges linearly to a limit t^* and that for i sufficiently large, $(t_{i+1}-t^*)(t_i-t^*)>0$, the sequence $\{\widehat{t}_i\}_{i=0}^{\infty}$, generated by Aitken's Δ^2 process converges to t^* faster in the sense that ([26, 27])

$$\lim_{i\to\infty}\frac{\widehat{t}_i-t^*}{t_i-t^*}=0.$$

Although Aitken's Δ^2 process does not generally converges quadratically, we note that when the terms $\{t_i\}_{i=0}^{\infty}$ come from a fixed point procedure, i.e., $t_{i+1}=f(t_i)$ for some function $f:\mathbb{R}\to\mathbb{R}$ which converges to a fixed point, the convergence rate can be shown to be quadratic [28]. This process is known as Steffensen's method.

The algorithm presented in this paper applies Aitken's Δ^2 process to the sequence formed by the estimates $\chi_0,\chi_1,\ldots$, obtained by Algorithm 1. Let now $k_i=2^i$ and $i+1\leq \mathtt{log}_2(n)$. The rate of convergence of the estimates $\{\chi_i\}_{i=0}^{i=\mathtt{log}_2(n)}$ is linear with a rate $\mu\in(0,1)$ if

$$\frac{|\chi_{i+1}-\mathtt{tr}(A)|}{|\chi_i-\mathtt{tr}(A)|}=\frac{\sum_{j=1}^{j=(n/2k_i)-1}\varepsilon_{2jk_i}}{\sum_{j=1}^{j=(n/k_i)-1}\varepsilon_{jk_i}}=\mu.$$

In practice, the sequence $\chi_0,\chi_1,\ldots$, can be accelerated even when the above ratios are only approximately equal.

Algorithm 2 Aitken's Δ^2 accelerated trace estimator

```
1. Set $\chi_{-1}=0$, $\chi_0=\mathbf{1}^T\mathbf{A}\mathbf{1}$, $\chi_1=\mathtt{tr}(\mathbf{Z}_2^T\mathbf{A}\mathbf{Z}_2)$, and
$i=-1$. Set a tolerance $\epsilon\in(0,1)$.
do
    2. Set $i=i+1,\ k_{i+2}=2^{i+2}$
    3. Compute $\chi_{i+2}=\mathtt{tr}(\mathbf{Z}_{k_{i+2}}^T\mathbf{A}\mathbf{Z}_{k_{i+2}})$
    4. Set $\widehat{\chi}_i=\chi_i-\frac{(\chi_{i+1}-\chi_i)^2}{(\chi_{i+2}-\chi_{i+1})-(\chi_{i+1}-\chi_i)}$
while $|\widehat{\chi}_i-\widehat{\chi}_{i-1}|>\epsilon\widehat{\chi}_i$, $\chi_i\neq\chi_{i-1}$, and $i\leq\mathtt{log}_2(n)-2$
```

The proposed method is listed in Algorithm 2. By construction, Algorithm 2 terminates after at most $\mathtt{log}_2(n)-2$ iterations. Compared to Algorithm 1, the cost to obtain the estimates $\widehat{\chi}_0,\widehat{\chi}_1,\ldots,\widehat{\chi}_{i-2}$ is roughly[4] equal to that of obtaining $\chi_0,\chi_1,\ldots,\chi_i$, and thus Algorithm 2 will be cost-effective compared to Algorithm 1 if and only if $\widehat{\chi}_{i-2}$ is a better approximation of $\mathtt{tr}(A)$ compared to χ_i. While Aitken's Δ^2 process breaks down if the sequence of first differences has a repeating term, this is not the case with Algorithm 2 since it terminates as soon as successive probing estimates χ_i and χ_{i+1} become numerically equal. Note also that the trace estimates $\widehat{\chi}_0,\widehat{\chi}_1,\ldots$, produced by Algorithm 2 can be further improved by applying Aitken's Δ^2 process *repeatedly*. In the next section we illustrate the effects of this idea.

4. NUMERICAL EXPERIMENTS

In this section we confirm the acceleration of probing trace estimators by Aitken's Δ^2 process on covariance matrices with a varying degree of feature correlation. Our experiments are conducted in a Matlab environment in 64-bit arithmetic, on a single core of a computing system equipped with a 2.3 GHz Quad-Core Intel Core i9 processor and 64 GB of system memory.

We compare four different methods: "*a*)" Algorithm 1, "*b*)" MC trace estimator with Rademacher vector variables, "*c*)" Algorithm 2, and "*d*)" a repeated application of Algorithm 2 to the sequence $\widehat{\chi}_i$ (double Aitken). The MC trace estimator provides estimates of the form $\mathtt{tr}(\mathbf{A})\approx\frac{1}{k}\sum_{j=1}^{j=k}\boldsymbol{z}_j^T\mathbf{A}\boldsymbol{z}_j$, where the entries of $\boldsymbol{z}_j\in\mathbb{R}^n$ are ±1 with equal probability. Since the main computational cost of all four approaches listed above stems from the evaluation of MV products $\mathbf{A}\boldsymbol{z}_j$, we will consider a method more efficient than another when it requires fewer MV products with matrix $\mathbf{A}$ to achieve the same accuracy. Moreover, the size of our test matrices will be set equal to $n=2,048$.

Our first set of experiments considers the following class of model covariance matrices:

$$\mathbf{A}_{ij}=\begin{cases}1 & \text{if } i==j\\ 1/|i-j|^{\theta} & \text{if } i\neq j\end{cases}, \tag{2}$$

where $\theta\in\mathbb{N}$ controls the correlation of the feature space. In particular, higher values of θ imply less correlation among the variables and lead to stronger decay in the off-diagonal entries of the model covariance matrix. Figure 1 plots the relative error achieved by the schemes "*a*)"-"*d*)" as the number of sample vectors 2^i increases and $\theta=\{1,2,3\}$. A few remarks are following. First, the accuracy of the trace estimates $\chi_0,\chi_1,\ldots$, returned by the probing technique listed in Algorithm 1 improves as the value of 2^i increases *(solid "□" line)*. The pair of integers listed on top of each entry denotes the value of 2^i required to compute the corresponding approximation (left integer) as well as the cumulative number $2^0+2^1+\ldots 2^i$ of MV

[4] The additional work needed to compute $\widehat{\chi}_0,\widehat{\chi}_1,\ldots,\widehat{\chi}_{i-2}$ requires only the calculation of four differences, one multiplication, and one division between scalars.

products up to this point (right integer). Second, Hutchinson's MC trace estimator *(solid "$\triangledown$" line)* provides a better estimate of $\mathtt{tr}(\mathbf{A})$ for low values of $k_i = 2^i$, as expected for matrices $\mathbf{A}$ which are approximately diagonally dominant. However, it converges slowly as k_i increases. The convergence rate of Algorithm 1 can be accelerated by applying Aitken's Δ^2 process to the sequence $\{\chi_i\}$, as discussed in Algorithm 2. This option is denoted as "Algorithm 2 (a)" *(dashed "$\bigcirc$" line)*. In addition, Aitken's Δ^2 process can be also applied on the sequence $\{\widehat{\chi}_i\}$, which is equivalent to double Aitken and denoted as "Algorithm 2 (b)" *(dashed "$\triangle$" line)*. Algorithm 2 is the most efficient scheme, especially when the double-Aitken variant is exploited. For example, when $\theta = 3$, Aitken's Δ^2 process can provide up to five orders of magnitude improvement for as low as $k_i = 16$.

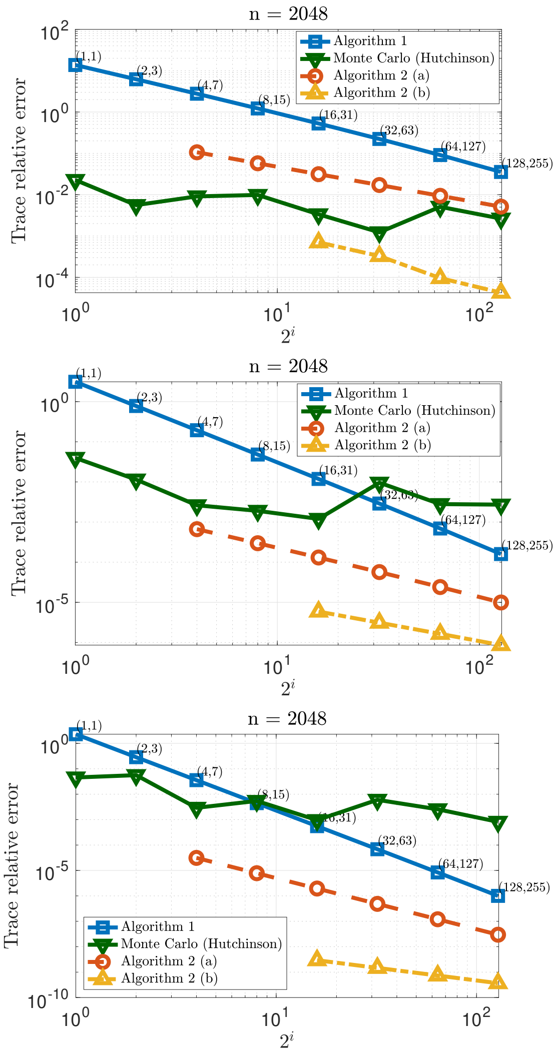

Fig. 1. Model covariance matrix (see (2)). Top, center, and bottom figures correspond to $\theta = 1$, $\theta = 2$ and $\theta = 3$, respectively.

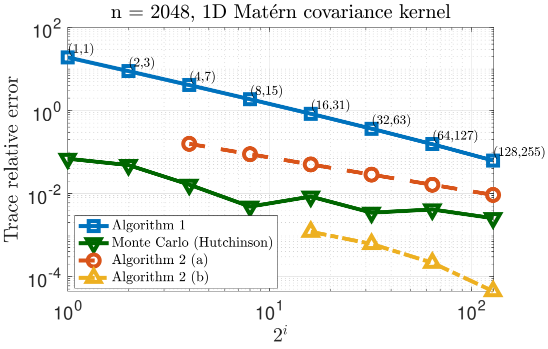

Fig. 2. Matérn covariance kernel (see (3)).

Our second class of test matrices originates from the Matérn covariance kernel [29]:

$$\phi(\boldsymbol{x}) = \frac{(\sqrt{2\nu}\boldsymbol{x}/\ell)^{\nu} K_{\nu}(\sqrt{2\nu}\boldsymbol{x}/\ell)^{\nu}}{2^{\nu-1}\Gamma(\nu)}, \tag{3}$$

where $\Gamma(.)$ is the Gamma function, $K_\nu(\cdot)$ is the modified Bessel function of the second kind of order ν, and ν is a positive scalar that controls the smoothness of the covariance kernel. The variable ℓ denotes the characteristic length scale. Figure 2 plots the same quantities as in Figure 1 for a Matérn covariance kernel discretized on a 1D grid with $\ell = 7$ and $\nu = 0.1$. Similarly to the results for the model covariance matrices, Algorithm 2 improves the probing trace estimator and can compete with the MC estimator once a sufficient number of samples is exploited.

5. CONCLUSION

This paper presented an algorithm to estimate the trace of symmetric matrices by applying Aitken's Δ^2 process to trace estimations produced using the probing technique. Numerical experiments performed on covariance matrix kernels of locally correlated features demonstrate competitive performance versus probing and Hutchinson's randomized trace estimator. As part of future work we plan to extend our work towards applying Aitken's Δ^2 process to compute the entire diagonal of a matrix function $f(\mathbf{A})$ where $f(.)$ is analytic inside the spectrum of $\mathbf{A}$. One particular application is the computation of vertex centralities and top-N graph recommendation. In this setting, determining the top-N most influential nodes of a network corresponds to estimating the main diagonal of matrix functions, e.g., $e^{\mathbf{A}}$ and $(\mathbf{I} - \alpha\mathbf{A})^{-1}$, where $\mathbf{A}$ denotes the adjacency matrix of the input undirected graph.

6. REFERENCES

[1] Vassilis Kalantzis, Constantine Bekas, Alessandro Curioni, and Efstratios Gallopoulos, “Accelerating data uncertainty quantification by solving linear systems with multiple right-hand sides,” *Numerical Algorithms*, vol. 62, no. 4, pp. 637–653, 2013.

[2] Ernesto Estrada, “Characterization of 3D molecular structure,” *Chemical Physics Letters*, vol. 319, no. 5-6, pp. 713–718, 2000.

[3] Eugenia-Maria Kontopoulou, Gregory-Paul Dexter, Wojciech Szpankowski, Ananth Grama, and Petros Drineas, “Randomized linear algebra approaches to estimate the von neumann entropy of density matrices,” *IEEE Transactions on Information Theory*, 2020.

[4] Andreas Stathopoulos, Jesse Laeuchli, and Kostas Orginos, “Hierarchical probing for estimating the trace of the matrix inverse on toroidal lattices,” *SIAM Journal on Scientific Computing*, vol. 35, no. 5, pp. S299–S322, 2013.

[5] Vassilis Kalantzis, A Cristiano I Malossi, Costas Bekas, Alessandro Curioni, Efstratios Gallopoulos, and Yousef Saad, “A scalable iterative dense linear system solver for multiple right-hand sides in data analytics,” *Parallel Computing*, vol. 74, pp. 136–153, 2018.

[6] Ernesto Estrada and Juan A. Rodríguez-Velázquez, “Spectral measures of bipartivity in complex networks,” *Phys. Rev. E*, vol. 72, pp. 046105, Oct 2005.

[7] Ernesto Estrada and Naomichi Hatano, “Statistical-mechanical approach to subgraph centrality in complex networks,” *Chemical Physics Letters*, vol. 439, no. 1-3, pp. 247–251, 2007.

[8] Lingfei Wu, Jesse Laeuchli, Vassilis Kalantzis, Andreas Stathopoulos, and Efstratios Gallopoulos, “Estimating the trace of the matrix inverse by interpolating from the diagonal of an approximate inverse,” *Journal of Computational Physics*, vol. 326, pp. 828–844, 2016.

[9] Jesse Laeuchli and Andreas Stathopoulos, “Extending hierarchical probing for computing the trace of matrix inverses,” *SIAM Journal on Scientific Computing*, vol. 42, no. 3, pp. A1459–A1485, 2020.

[10] Michele Benzi, Paola Boito, and Nader Razouk, “Decay properties of spectral projectors with applications to electronic structure,” *SIAM review*, vol. 55, no. 1, pp. 3–64, 2013.

[11] Haim Avron and Sivan Toledo, “Randomized algorithms for estimating the trace of an implicit symmetric positive semi-definite matrix,” *Journal of the ACM (JACM)*, vol. 58, no. 2, pp. 1–34, 2011.

[12] Andrew Brock, Jeff Donahue, and Karen Simonyan, “Large scale gan training for high fidelity natural image synthesis,” *arXiv preprint arXiv:1809.11096*, 2018.

[13] Konstantin Shmelkov, Cordelia Schmid, and Karteek Alahari, “How good is my gan?,” in *Proceedings of the European conference on computer vision (ECCV)*, 2018, pp. 213–229.

[14] Mihai Anitescu, Jie Chen, and Lei Wang, “A matrix-free approach for solving the parametric gaussian process maximum likelihood problem,” *SIAM Journal on Scientific Computing*, vol. 34, no. 1, pp. A240–A262, 2012.

[15] Michael F Hutchinson, “A stochastic estimator of the trace of the influence matrix for Laplacian smoothing splines,” *Communications in Statistics-Simulation and Computation*, vol. 18, no. 3, pp. 1059–1076, 1989.

[16] Jok M Tang and Yousef Saad, “A probing method for computing the diagonal of a matrix inverse,” *Numerical Linear Algebra with Applications*, vol. 19, no. 3, pp. 485–501, 2012.

[17] Jok M Tang and Yousef Saad, “Domain-decomposition-type methods for computing the diagonal of a matrix inverse,” *SIAM Journal on Scientific Computing*, vol. 33, no. 5, pp. 2823–2847, 2011.

[18] A. C. Aitken, “On Bernoulli’s numerical solution of algebraic equations,” *Proceedings of the Royal Society of Edinburgh*, vol. 46, pp. 289–305, 1927.

[19] Claude Brezinski and M Redivo Zaglia, *Extrapolation methods: theory and practice*, Elsevier, 2013.

[20] Paraskevi Fika, “Extrapolation methods for estimating the trace of the matrix inverse,” in *Modern Discrete Mathematics and Analysis*, pp. 173–185. Springer, 2018.

[21] Paraskevi Fika and Marilena Mitrouli, “Aitken’s method for estimating bilinear forms arising in applications,” *Calcolo*, vol. 54, no. 1, pp. 455–470, 2017.

[22] Andreas Frommer, Claudia Schimmel, and Marcel Schweitzer, “Analysis of probing techniques for sparse approximation and trace estimation of decaying matrix functions,” *arXiv preprint arXiv:2009.01589*, 2020.

[23] Sepandar D Kamvar, Taher H Haveliwala, Christopher D Manning, and Gene H Golub, “Extrapolation methods for accelerating pagerank computations,” in *Proceedings of the 12th international conference on World Wide Web*, 2003, pp. 261–270.

[24] Peter Wynn, “On a device for computing the e m (s n) transformation,” *Mathematical Tables and Other Aids to Computation*, pp. 91–96, 1956.

[25] C Brezinski, “Convergence acceleration during the 20th century,” *Numerical Analysis: Historical Developments in the 20th Century*, vol. 113, 2001.

[26] D Kincaid and W Cheney, *Numerical Analysis Mathematics of Scientific Computing*, Pacific Grove, CA: Brooks/-Cole, 1990.

[27] I Faragó, “Some notes on the sequence acceleration,” *J Appl Computat Math*, vol. 5, no. 303, pp. 2, 2016.

[28] I Păvăloiu, “Approximation of the roots of equations by aitken-steffensen-type monotonic sequences,” *Calcolo*, vol. 32, no. 1, pp. 69–82, 1995.

[29] Christopher KI Williams and Carl Edward Rasmussen, *Gaussian processes for machine learning*, vol. 2, MIT press Cambridge, MA, 2006.