Improved Stochastic Trace Estimation using Mutually Unbiased Bases

Jack Fitzsimons[1] **Michael Osborne**[1] **Stephen Roberts**[1] **Joseph Fitzsimons**[2,3]

[1] Information Engineering, University of Oxford, UK
[2] Singapore University of Technology and Design, Singapore
[3] Centre for Quantum Technologies, National University of Singapore, Singapore

Abstract

The paper begins by introducing the definition and construction of mutually unbiased bases, which are a widely used concept in quantum information processing but have received little to no attention in the machine learning and statistics literature. We demonstrate their usefulness by using them to create a new sampling technique which offers an improvement on the previously well established bounds of stochastic trace estimation. This approach offers a new state of the art single shot sampling variance while requiring $\mathcal{O}(\log(n))$ random bits for $\mathbf{x} \in \mathbb{R}^n$ which significantly improves on traditional methods such as fixed basis methods, Hutchinson's and Gaussian estimators in terms of the number of random bits required and worst case sample variance.

1 INTRODUCTION

Function space representations and transformations are at the heart of many machine learning techniques. For example, the relationship between computational space and Fourier space arrises throughout machine learning literature, from classic shift invariant filters studied in image processing through to modern techniques for kernel approximation such as random Fourier features [1]. The power of random Fourier features, for example, is introduced by the unbiased relationship of the computational and Fourier bases, that is to say that a Dirac-delta distribution in one basis is represented with uniformly distributed mass in the other.

In this work we take advantage of not only pairs of mutually unbiased bases, but entire sets of them. We believe that the application of mutually unbiased bases has potential to improve kernel matrix approximation and feature learning. To exemplify their applicability, we demonstrate their ability to create a novel sampling method which improves upon the well established error bounds of stochastic trace estimation.

The problem of stochastic trace estimation is relevant to a range of problems from physics and applied mathematics such as electronic structure calculations [2], seismic waveform inversion [3], discretized parameter estimation problems with PDEs as constraints [4] and approximating the log determinant of symmetric positive semi-definite matrices [5]. Machine learning, in particular, is a research domain which has many uses for stochastic trace estimation. They have been used efficiently by Generalised Cross Validation (GCV) in discretized iterative methods for fitting Laplacian smoothing splines to very large datasets [6], computing the number of triangles in a graph [7, 8], string pattern matching [9, 10] and training Gaussian Processes using score functions [11]. Motivated by accelerating Gaussian graphical models, Markov random fields, variational methods and Bregman divergences, work based on stochastic trace estimation has also been developed to improve the computational efficiency of log determinant calculations [12].

Stochastic trace estimation endeavours to choose n-dimensional vectors $\mathbf{x}$ such that the expectation of $\mathbf{x}^T A \mathbf{x}$ is equal to the trace of the implicit symmetrical positive semi definite matrix $A \in \mathbb{R}^{n \times n}$. It can be seen that many sampling policies satisfy this condition. Due to this, several metrics are used in order to choose a sampling policy such as the single shot sampling variance, the number of samples to achieve a (ε, δ)-approximation and the number of random bits required to create $\mathbf{x}$ [13]. This last metric is motivated in part by the relatively long timescales for hardware random number generation, and concerns about parallelising pseudo-random number generators.

In this work we propose a new stochastic trace estimator based on mutually unbiased bases (MUBs) [14], and

quantify the single shot sampling variance of the proposed MUBs sampling method and its corresponding required number of random bits. We will refer to methods which sample from a fixed set of basis functions as being fixed basis sampling methods. For example, we can randomly sample the diagonal values of the matrix A by sampling $\mathbf{x}$ from the set of columns which form the identity matrix. This is referred to as the unit vector estimator in the literature [13]. Other similar methods sample from the columns Discrete Fourier Transform (DFT), the Discrete Hartley Transform (DHT), the Discrete Cosine Transform (DCT) or a Hadamard matrix. We prove that sampling from the set of mutually unbiased bases significantly reduces this single shot sample variance, in particular in the worst case bound.

The paper is laid out as follows: Section 2 gives a brief introduction to mutually unbiased basis and their construction, Section 3 describes our novel approach of using mutually unbiased bases for trace estimation and Section 3.2 gives a rigorous analysis of of the new estimator. Section 4 compares the proposed MUBs estimator to established approaches both in terms of the analytic expectation of sample variance and as applied to synthetic and real data. The tasks of counting the number of triangles in a graph and of estimating the log determinant of kernel matrices are considered as an example application.

2 MUTUALLY UNBIASED BASES

Linear algebra has found application in a diverse range of fields, with each field drawing from a common set of tools. However, occasionally, techniques developed in one field do not become well known outside of that community, despite the potential for wider use. In this work, we will make extensive use of mutually unbiased bases, sets of bases that arise from physical considerations in the context of quantum mechanics [14] and which have been extensively exploited within the quantum information community [15]. In quantum mechanics, physical states are represented as vectors in a complex vector space, and the simplest form of measurement projects the state onto one of the vectors from some fixed orthonormal basis for the space, with the probability for a particular outcome given by the square of the length of the projection onto the corresponding basis vector [1]. In such a setting, it is natural to ask about the existence of pairs or sets of measurements where the outcome of one measurement reveals nothing about the outcome of another measurement, and effectively erases any information about the outcome had the alternate measurement instead been performed. As each measurement corresponds to a particular basis, such a requirement implies that the absolute value of the overlap between pairs of vectors drawn from bases corresponding to different measurements be constant. This leads directly to the concept of mutually unbiased bases (MUBs).

A set of orthonormal bases $\{B_1,\ldots,B_n\}$ are said to be mutually unbiased if for all choices of i and j, such that $i \neq j$, and for every $\mathbf{u} \in B_i$ and every $\mathbf{v} \in B_j$, $|\mathbf{u}^\dagger \mathbf{v}| = \frac{1}{\sqrt{n}}$, where n is the dimension of the space. While for real vector spaces the number of mutually unbiased bases has a complicated relationship with the dimensionality [17], for complex vector spaces the number of mutually unbiased bases is known to be exactly $n+1$ when n is either a prime or an integer power of a prime [18]. Furthermore, a number of constructions are known for finding such bases [18]. When n is neither prime nor a power of a prime, the number of mutually unbiased bases remains open, even for the case of $n = 6$ [19], but is known to be at least $p_1^{d_1} + 1$, where $n = \prod_i p_i^{d_i}$ and p_i are prime numbers such that $p_i < p_{i+1}$ for all i.

One practical method for constructing MUBs is to use the unitary operators method with finite fields [20], which is effective when the dimensionality of the space is either prime or a prime power. For conciseness, we will outline the procedure for only the prime dimensionality case but note that any integer dimensional space is at most bounded by two times its closest prime power dimension which adds a constant cost to the memory and runtime performance. First, let us construct the matrix X as the identity matrix with the columns shifted one to the left creating the form,

$$X = \begin{bmatrix} 0 & 1 & 0 & \ldots & 0 \\ 0 & 0 & 1 & \ldots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \ldots & 1 \\ 1 & 0 & 0 & \ldots & 0 \end{bmatrix}$$

and letting Z be a diagonal matrix with elements set to the roots of unity, $Z_{k,k} = \exp\left(\frac{2k\pi i}{n}\right)$. Given these two matrices a set of mutually unbiased bases are found as the eigenvectors of the matrices,

$$X, Z, XZ, XZ^2, \ldots, XZ^{n-1}.$$

At first glance it may appear that the computational cost of constructing vectors from these bases is $\mathcal{O}(d^3)$ due to the cost decomposing these matrices in $\mathbb{C}^{n\times n}$, however, under more scrutiny we can see that X is a circulant permutation matrix and as such its eigenvectors are equal to $U_{k,j} = \frac{1}{\sqrt{n}} \exp\left(\frac{jk2\pi i}{n}\right)$ irrespective of the dimensionality,

[1] For a more comprehensive introduction to the mathematics of quantum mechanics in finite-dimensional systems, we refer the reader to [16]

where j indexes the elements of the eigenvector and v indexes which eigenvector is under consideration.

As the elements the diagonal matrix Z are the roots of unity in ascending order, it can be seen that $\exp\left(\frac{2\pi i}{n}\right)^k Z^k X = Q^k X = XZ^k$, where Q is some matrix of the same form as Z but with a shift of phase of the non-zero elements. As such, by writing the eigenbasis of $X = U^{-1}\Sigma U$ we can derive the eigenbasis of XZ^k for arbitrary value k with eigen decomposition $XZ^k = \hat{U}^{-1}\hat{\Sigma}\hat{U}$,

$$\begin{aligned} XZ^k &= U^\dagger \Sigma U Z^k \\ &= Q^k U^{-1}\Sigma U \end{aligned}$$

Next, we pull $Q^{\frac{k}{2}}$ and $Z^{\frac{k}{2}}$ through the eigenvectors by observing that we can transform the eigenvectors as $\Sigma = Z^{\frac{1}{2}}\hat{\Sigma}Q^{\frac{k}{2}}$,

$$\begin{aligned} XZ^k &= Q^{\frac{k}{2}}U^{-1}\Sigma U Z^{\frac{k}{2}} \\ &= Q^{\frac{k}{2}}U^{-1}Z^{\frac{1}{2}}\hat{\Sigma}Q^{\frac{k}{2}}UZ^{\frac{k}{2}} \\ &= \hat{U}^{-1}\hat{\Sigma}\hat{U} \end{aligned}$$

where $\hat{U} = Q^{-\frac{k}{2}}Z^{\frac{k}{2}}U$ and hence the $\hat{U}_{i,j} = \frac{1}{\sqrt{n}}\exp\left(\frac{2\pi i}{n}(jv + \frac{(j+1)(j+2)}{2}k)\right)$ using the same indexing as before.

As a result, we can simply use the following procedure to sample the vector $\mathbf{x}$ in linear computational time and memory:

1 Choose k and v, representing the basis and the vector to select respectively, uniformly at random.

2 If $k = 0$, then we select the vector v from the computational basis, that is to say the columns of the identity matrix.

3 Else, let $x_j = \frac{1}{\sqrt{n}}\exp\left(\frac{2\pi i}{n}(jv + \frac{(j+1)(j+2)}{2}k\right)$

3 TRACE ESTIMATORS

In order to estimate the trace of a $n \times n$ positive semi-definite matrix A from a single call to an oracle for $\mathbf{x}^\dagger A\mathbf{x}$, we consider four strategies:

- **Fixed basis estimator:** For a fixed orthonormal basis B, choose $\mathbf{x}$ uniformly at random from the elements of B. The trace is then estimated to be $n\mathbf{x}^\dagger A\mathbf{x}$.
- **Mutually unbiased bases (MUBs) estimator:** For a fixed choice of a set of b mutually unbiased bases $\mathbb{B} = \{B_1, ..., B_b\}$, choose B uniformly at random from $\mathbb{B}$ and then choose $\mathbf{x}$ uniformly at random from the elements of B. Here b is taken to be the maximum number of mutually unbiased bases for a complex vector space of dimension n. As in the fixed basis strategy, the trace is then estimated to be $n\mathbf{x}^\dagger A\mathbf{x}$.
- **Hutchinson's estimator:** Randomly choose the elements of $\mathbf{x}$ independently and identically distributed from a Rademacher distribution $\left(Pr(x_i = \pm 1) = \frac{1}{2}\right)$. The trace is then estimated to be $\mathbf{x}^\dagger A\mathbf{x}$.
- **Gaussian estimator:** Randomly choose the elements of $\mathbf{x}$ independently and identically distributed from a zero mean unit variance Gaussian distribution. The trace is then estimated to be $\mathbf{x}^\dagger A\mathbf{x}$.

The first strategy is a generic formulation of approaches which sample vectors from a fixed orthogonal basis, the most efficient sampling method in terms of the number of random bits required in the literature [13], while the second strategy is novel and represents our main contribution. Both strategies have similar randomness requirements: In the first strategy at least $\lceil \log_2(n) \rceil$ random bits are necessary to ensure the possibility of choosing every element of B. In the second strategy, an identical number of random bits is necessary to choose $\mathbf{x}$ for a fixed B, and $\lceil \log_2(b) \rceil$ random bits are necessary to choose B. Note that an upper bound on the number of mutually unbiased bases is one greater than dimensionality of the space, and this bound is saturated for spaces where the dimensionality is prime or an integer power of a prime, i.e. $b \leq n+1$. Thus the number of random bits necessary to implement these strategies differs by a factor of approximately two. The third and forth strategies significantly outperform the fixed basis estimator in terms of single-shot variance, at the cost of a dramatic increase in the amount of randomness required, and have been extensively studied in the literature [13, 21, 22]. For conciseness we will not repeat the analysis of these methods in this paper but will compare the fixed basis estimator and MUBs estimator to them in Table 4.1.

3.1 ANALYSIS OF FIXED BASIS ESTIMATOR

We first analyse the worst case variance of the fixed base estimator. In this analysis and the analysis for the MUBs estimator which follows, we make no assumption on A and consider the worst case variance.

We begin from the definition of the variance of the estimator for a single query. Let X be a random variable

such that $X = \mathbf{x}^\dagger A \mathbf{x}$, where $\mathbf{x}$ is chosen according to the fixed basis strategy. Then

$$\mathrm{Var(X)} = E(X^2) - E(X)^2, \tag{1}$$

where $E(\cdot)$ denotes the expectation value of the argument. We compute this term by term. First

$$E(X) = \frac{1}{n}\sum_{\mathbf{x}\in B} \mathbf{x}^\dagger A \mathbf{x} = \frac{\mathrm{Tr}(A)}{n}.$$

where $n = \dim A$, and hence the second term in Eq. 1 is equal to $\frac{\mathrm{Tr}(A)^2}{n^2}$. Turning to the first term,

$$\begin{aligned} E(X^2) &= \frac{1}{n}\sum_{\mathbf{x}\in B} \left(\mathbf{x}^\dagger A \mathbf{x}\right)^2 \\ &= \frac{1}{n}\sum_{i=1}^{n} M_{ii}^2, \end{aligned}$$

where $M = UAU^\dagger$ for some fixed unitary matrix U such that $U^\dagger \mathbf{x}$ is a vector in the standard basis for all $x \in B$, and M_{ii} is the ith entry on the main diagonal of M. The variance for the fixed basis estimator is then given by $V_{\mathrm{fixed}} = n\sum_{i=1}^{n} M_{ii}^2 - \mathrm{Tr}(A)^2$. The worst case occurs when the value of $\sum_{i=1}^{n} M_{ii}^2$ is maximized for fixed trace of A (and hence M), and so the worst case single shot variance for the fixed basis estimator is $V_{fixed}^{\mathrm{worst}} = (n-1)\mathrm{Tr}(A)^2$.

3.2 ANALYSIS OF MUBS ESTIMATOR

We now turn to analysis of the MUBs estimator. We assume that n is either prime or a prime raised to some integer power. In this case, it has been established that $b = n+1$ [18]. The variance is defined as in Eq. 1, except that X is defined in terms of $\mathbf{x}$ chosen according to the MUBs strategy. Again, we analyse the individual terms making up the variance. We begin with

$$E(X) = \frac{1}{nb}\sum_{B\in\mathbb{B}}\sum_{\mathbf{x}\in B} \mathbf{x}^\dagger A \mathbf{x} = \frac{\mathrm{Tr}(A)}{n}.$$

and hence the second term in the variance is the same as for the fixed basis estimator. Analysing the first term is, however, more difficult. We begin with the observation that $E(X^2)$ can be expressed in terms of the trace of the Kronecker product of two matrices, as follows

$$\begin{aligned} E(X^2) &= \frac{1}{nb}\sum_{B\in\mathbb{B}}\sum_{\mathbf{x}\in B} \left(\mathbf{x}^\dagger A \mathbf{x}\right)^2 \\ &= \frac{1}{nb}\sum_{B\in\mathbb{B}}\sum_{\mathbf{x}\in B} \mathrm{Tr}\left((\mathbf{x}\mathbf{x}^\dagger A)^{\otimes 2}\right). \end{aligned}$$

Moving the summations inside the equation we obtain

$$\begin{aligned} E(X^2) &= \frac{1}{nb}\mathrm{Tr}\left(\sum_{B\in\mathbb{B}}\sum_{\mathbf{x}\in B} \left(\mathbf{x}\mathbf{x}^\dagger\right)^{\otimes 2} A^{\otimes 2}\right) \\ &= \frac{2}{nb}\mathrm{Tr}\left(PA^{\otimes 2}\right), \end{aligned} \tag{2}$$

where $P = \frac{1}{2}\sum_{B\in\mathbb{B}}\sum_{\mathbf{x}\in B}\left(\mathbf{x}\mathbf{x}^\dagger\right)^{\otimes 2}$.

While this form of P may appear intimidating, we now prove that P is in fact a projector with each eigenvalue being either 0 or 1. We prove this indirectly, first by showing that P has rank at most $n(n+1)/2$, and then using the relationship between the traces of P and P^2 to conclude that the non-zero $n(n+1)/2$ eigenvalues are equal to unity. Any vector of the form $\mathbf{w} = \mathbf{u}\otimes\mathbf{v} - \mathbf{u}\otimes\mathbf{v}$ for $\mathbf{u},\mathbf{v} \in B_1$ trivially satisfies $P\mathbf{w} = \mathbf{0}$. Since such vectors form a basis for a subspace of dimension $n(n-1)/2$, we conclude that $\mathrm{rank}(P) \leq n^2 - n(n-1)/2 = n(n+1)/2$. Turning now to the issue of trace, we have

$$\begin{aligned} \mathrm{Tr}(P) &= \mathrm{Tr}\left(\frac{1}{2}\sum_{B\in\mathbb{B}}\sum_{\mathbf{x}\in B}\left(\mathbf{x}\mathbf{x}^\dagger\right)^{\otimes 2}\right) \\ &= \frac{1}{2}\sum_{B\in\mathbb{B}}\sum_{\mathbf{x}\in B}\left(\mathbf{x}^\dagger\mathbf{x}\right)^2 \\ &= \frac{nb}{2}. \end{aligned}$$

We can similarly compute the trace of P^2 to obtain

$$\begin{aligned} \mathrm{Tr}(P^2) &= \mathrm{Tr}\left(\frac{1}{4}\sum_{B,B'\in\mathbb{B}}\sum_{\mathbf{x}\in B}\sum_{\mathbf{y}\in B'}\left(\mathbf{x}\mathbf{x}^\dagger\right)^{\otimes 2}\left(\mathbf{y}\mathbf{y}^\dagger\right)^{\otimes 2}\right) \\ &= \frac{1}{4}\sum_{B,B'\in\mathbb{B}}\sum_{\mathbf{x}\in B}\sum_{\mathbf{y}\in B'}\left|\mathbf{x}^\dagger\mathbf{y}\right|^4 \\ &= \frac{nb}{4} + \frac{n^2 b(b-1)}{4n^2} \\ &= \frac{b(n+b-1)}{4}. \end{aligned}$$

Notice that this implies that $\mathrm{Tr}(P) = \mathrm{Tr}(P^2)$ for dimensions which are prime or integer powers of a prime, since in such cases $b = n+1$. This implies that the eigenvalues on the non-zero subspace minimize the sum of their squares for a fixed sum, and since P is positive semi-definite, we can conclude that each non-zero eigenvalue must be equal to unity.

Returning to the calculation of variance, we then have

$$\begin{aligned} E(X^2) &\leq \frac{2}{nb}\mathrm{Tr}\left(A^{\otimes 2}\right) \\ &= \frac{2}{nb}\mathrm{Tr}\left(A\right)^2, \end{aligned}$$

and hence

$$\mathrm{Var}(X) \leq \left(\frac{2}{nb} - \frac{1}{n^2}\right)\mathrm{Tr}\left(M\right)^2 \leq \frac{\mathrm{Tr}\left(A\right)^2}{n^2}. \tag{3}$$

This implies that the variance on the estimate of $\mathrm{Tr}(A)$ is bounded from above by $\mathrm{Tr}(A)^2$. It is, in fact, possible to compute the variance exactly from Eq. 2 by observing

that M is the projector onto the symmetric subspace when n is an integer power of a prime. That is to say, for any vector $\mathbf{u}$ and any vector $\mathbf{v}$ orthogonal to $\mathbf{u}$, the vectors $\mathbf{u}\otimes\mathbf{v}+\mathbf{v}\otimes\mathbf{u}$, $\mathbf{u}\otimes\mathbf{u}$ and $\mathbf{v}\otimes\mathbf{v}$ are in the $+1$ eigenspace of M, whereas the vector $\mathbf{u}\otimes\mathbf{v}-\mathbf{v}\otimes\mathbf{u}$ is in the null space of M. Thus we can compute the exact variance of the MUBs estimator, using the spectral decomposition $A = \sum_i \lambda_i \mathbf{u}_i \mathbf{u}_i^\dagger$ as

$$\begin{aligned}
\mathrm{V}_{\mathrm{MUBs}} &= \frac{2n}{n+1}\mathrm{Tr}\left(PA^{\otimes 2}\right) - \mathrm{Tr}(A)^2 \\
&= \frac{2n}{n+1}\sum_{i=1}^{n}\sum_{j=1}^{n}\lambda_i\lambda_j \\
&\qquad \mathrm{Tr}\left(P(\mathbf{u}_i\otimes\mathbf{u}_j)(\mathbf{u}_i\otimes\mathbf{u}_j)^\dagger\right) - \mathrm{Tr}(A)^2 \\
&= \frac{2n}{n+1}\sum_{i=1}^{n}\left(\lambda_i^2 + \frac{1}{2}\sum_{j\neq i}\lambda_i\lambda_j\right) - \mathrm{Tr}(A)^2 \\
&= \frac{n}{n+1}\mathrm{Tr}(A^2) - \frac{1}{n+1}\mathrm{Tr}(A)^2.
\end{aligned}$$

Since for all positive semi-definite matrices A the value of $\mathrm{Tr}(A)^2$ is bounded from below by $\mathrm{Tr}(A^2)$, the single shot variance on the MUBs estimator is bounded by $\mathrm{V}_{\mathrm{MUBs}}^{\mathrm{worst}} = \frac{n-1}{n+1}\mathrm{Tr}(A^2)$ in the worst case, a significant improvement on the bound stemming from Eq. 3. The worst case single shot variance of the MUBs estimator is then at least a factor of $n+1$ better than that of any fixed basis estimator. Furthermore, the variance for the widely used Hutchinson estimator [21, 13], is given by $V_{\mathrm{H}} = 2\left(\mathrm{Tr}(A^2) - \sum_{i=1}^{n} A_{ii}^2\right)$. In the worst case, $\sum_{i=1}^{n} A_{ii}^2 = \frac{1}{n}\mathrm{Tr}(A^2)$, and hence the worst case single shot variance for Hutchinson estimator is $V_{\mathrm{H}}^{\mathrm{worst}} = \frac{2(n-1)}{n}\mathrm{Tr}(A^2)$. Thus, the MUBs estimator has better worst case performance than the Hutchinson estimator by a factor $\frac{2(n+1)}{n}$ which approaches 2 from above for large n.

4 RESULTS

4.1 THEORETICAL RESULTS

Table 1 compares the single shot variance, worst case single shot variance and randomness requirements of the trace estimators. As can be seen from the comparison the MUBs estimator has strictly smaller variance than either the Hutchinson or Gaussian methods, while requiring significantly less randomness to implement. Given the drastic reduction in randomness requirements, and the improved worst case performance, the MUBs estimator provides an attractive alternative to previous methods for estimating the trace of implicit matrices.

4.2 NUMERICAL RESULTS

4.2.1 Example Matrices

Before we demonstrate the use of the MUBs estimator on example applications we draw the readers attention to a situation where the traditional methods perform poorly. This occurs when the values of the matrix A are close to the ones matrix with a small proportion of the diagonal values much greater. The most extreme example being when this small proportion is only one element of the matrix. Due to the relationship between each of the unbiased bases this 'spikiness' only appears in one of the $n+1$ bases and hence the MUBs estimator appears very robust to the condition.

It is worth noting the reason we observe an order of magnitude improvement in this setting over the competing methods. The spikes matrix described can be written as the sum of two rank one matrices. Each of these matrices will perform very poorly for the unitary estimator in that basis but gets exactly the correct result in the n other mutually unbiased bases. Naturally as n becomes large and the number of samples utilised is relatively small, then we sample the exact result with high probability.

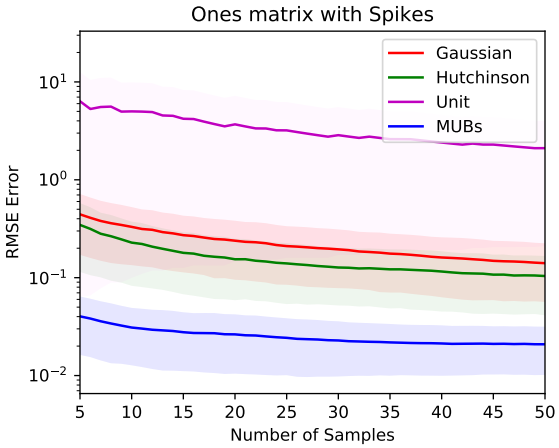

Figure 1: Convergence of the methods when estimating the trace of a 1000×1000 ones matrix with 1 diagonal element replaced with 1001. This 'spike' has little effect of the convergence of the MUBs estimator and hence the method vastly out performs the others. The experiment was run 500 times and the mean and standard deviation have been plotted for each method.

We can generalise this result to low rank matrices more broadly. Any given rank-m matrix can be written as the sum of m rank-1 matrices. Figure 2, demonstrates the convergence of of the stochastic trace estimators to rank-10 1000×1000 matrices. These were created by sam-

Estimator	V	V^{worst}	R
Fixed basis	$n\sum_{i=1}^{n} M_{ii}^2 - \text{Tr}(A)^2$	$(n-1)\text{Tr}(A)^2$	$\log_2(n)$
MUBs	$\frac{n}{n+1}\text{Tr}(A^2) - \frac{1}{n+1}\text{Tr}(A)^2$	$\frac{n-1}{n+1}\text{Tr}(A^2)$	$\log_2(n) + \log_2(n+1)$
Hutchinson [21]	$2\left(\text{Tr}(A^2) - \sum_{i=1}^{n} A_{ii}^2\right)$	$\frac{2(n-1)}{n}\text{Tr}(A^2)$	n
Gaussian [22]	$2\text{Tr}(A^2)$	$2\text{Tr}(A^2)$	∞ for exact; $\mathcal{O}(n)$ for fixed precision

Table 1: Comparison of single shot variance V, worst case single shot variance V^{worst} and number of random bits R required for commonly used trace estimators and the MUBs estimator.

pling 10 eigenvalues from a standard χ^2 distribution and sampling the first 10 eigenvectors of a Gaussian random matrix.

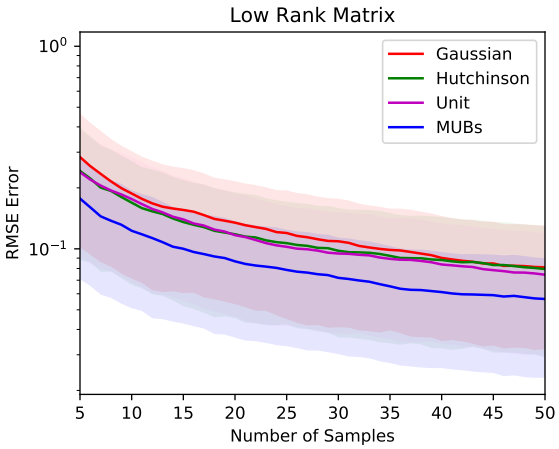

Figure 2: Convergence of the methods when estimating the trace of a 1000×1000 rank-10 matrix. The eigenvalues were sampled from a standard χ^2-distribution. As the rank of the matrix is only 1% of the dimensionality of the space we once again see substantially improved convergence rates. The experiment was run 30 times and the mean and standard deviation have been plotted for each method.

4.2.2 Counting Triangles in Graphs

As an example application we will consider counting the number of triangles in a graph. This is an important problem in a number of application domains such as identifying the number of 'friend of a friend' connections in a social network which is important for friendship suggestions [23, 24], identifying spam like behaviour [25] and even identifying thematic structures in the internet [26]. An efficient method to do this is the Trace Triangle algorithm [9]. The algorithm is based on a relationship between the adjacency matrix, A, and the number of triangles for an undirected graph, Δ_g,

$$\Delta_g = \frac{Tr(A^3)}{6}.$$

The trace of the adjacency matrix cubed can be sampled in $\mathcal{O}(n^2)$ per sample as opposed to being explicitly computed in $\mathcal{O}(n^3)$. We compared Gaussian, Hutchinson's, Unit and MUBs estimators performance at predicting the number of triangles for the graphs presented in Table 2 and the results of the experiment are presented in Figure 3. An efficient Python implementation for generating the MUBs sample vectors in $\mathcal{O}(n)$, is available at `www.github.com/OxfordMLRG/traceEst`. The MUBs estimator outperforms each of the classical methods in all of the experiments, as would be implied by the theory.

Dataset	Vertices	Edges	Triangles
Arxiv-HEP-th	27,240	341,923	1,478,735
CA-AstroPh	18,772	198,050	1,351,441
CA-GrQc	5,242	14,484	48,260
wiki-vote	7,115	100,689	608,389

Table 2: Datasets used for the comparison of stochastic trace estimation methods in the counting of triangles in graphs. All datasets can be found at `snap.stanford.edu/data`

4.2.3 Log Determinant of Covariance Matrix

Next, let us consider a common linear algebraic calculation required in the training of Gaussian processes, determinantal point processes and Gauss Markov random field modelling to name just a few applications, namely the log determinant of a kernel matrix.

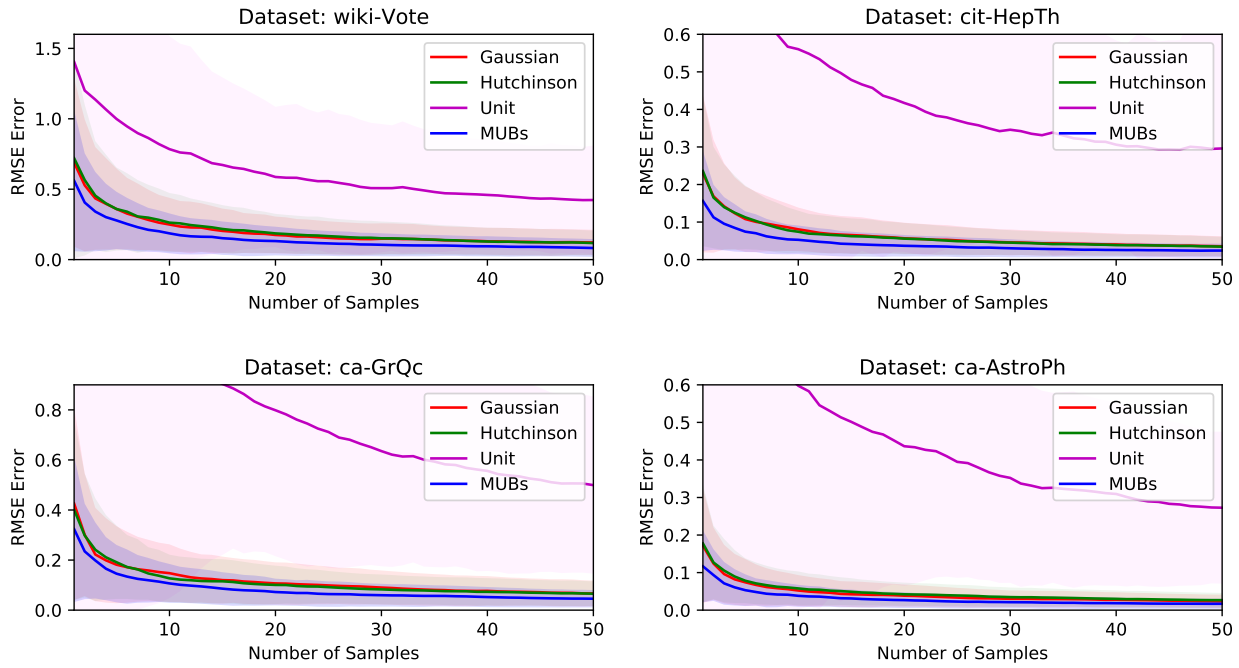

Figure 3: A comparison of the performance of the stochastic trace estimation methods on the four datasets. The experiments were performed 500 times each. The solid line indicated the empirical mean absolute relative error and the the surrounding transparent region indicates one empirical standard deviation of the 500 trials.

The use of stochastic trace estimation to approximate log determinant calculations of kernel matrices has been well studied [12, 27, 28] and a range of methods are feasible. Most notably, polynomial approaches such as truncated Taylor approximations and Chebyshev approximations [12, 29] have been applied, with the latter achieving consistently better results. The general concept relies on the fact that the trace of a matrix is simply the sum of its eigenvalues and the log determinant is the sum of the log of its eigenvalues. Stochastic trace estimation aids us in approximating the sum of the eigenvalues squared, cubed and so on which we can use in a polynomial approximation of the log function,

$$\log(x) \approx \sum_{j=0}^{m} c_j x^j \quad \rightarrow \quad \log(|K|) \approx \sum_{j=0}^{m} c_j \mathrm{Tr}(K^j)$$

where the constants c_j refer to the coefficients of the polynomial approximation. In practice, the trace of K^0 is simply the dimensionality of the matrix, K^1 is the trace of the explicit matrix and K^2 can be found as $\sum_{i,j} K_{i,j}$ due to the relationship between the matrix elements and the Frobenius norm. As such, the approximation error incurred is only due to the trace of the matrix raised to three and above.

In order to demonstrate the effect of improved stochastic trace estimation on log determinant estimations, we sampled 1000 points from a 5-dimensional hypercube uniformly at random. These points in turn formed a covariance matrix using an isotropic Gaussian kernel function. This aimed to emulate a realistic dataset which may be used by practitioners.

We used a order-6 Chebyshev polynomial approximation and recorded estimation errors of the relative root mean squared error (RMSE) for each power of the covariance matrix. These can be seen in Figure 4. Also plotted is the estimation error of the log determinant itself, as it compounds both the polynomial approximation error and the error due to the stochastic trace estimation. A fixed budget of 25 probing vectors was allowed for each of the approaches. As can be seen in the figure, the error incurred due to the stochastic trace estimation is non-negligible and for the higher order estimates the MUBs approach was achieving improved results in turns of both its expectation and standard error.

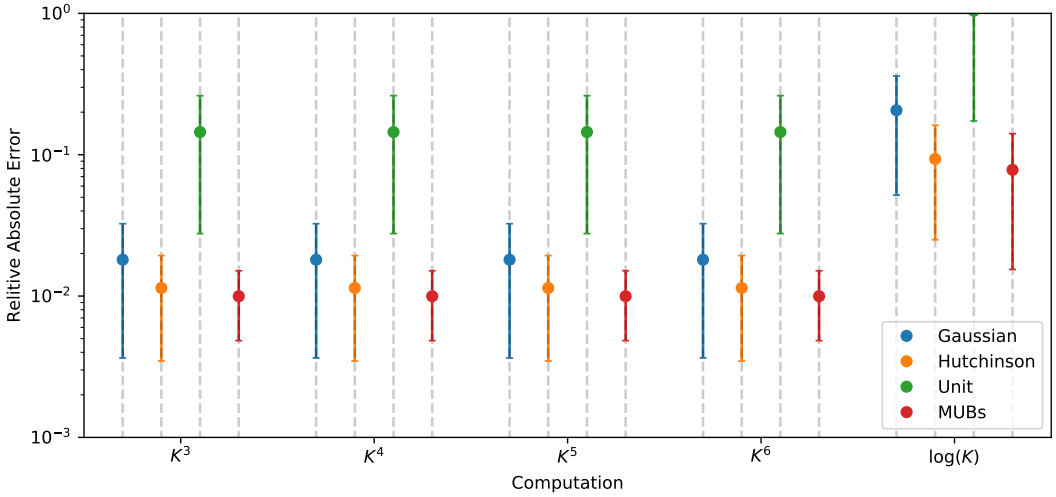

Figure 4: The performance of estimating the trace of K^3, K^4, K^5, K^6 and their combined result in the Chebyshev polynomial approximation of $\log(|K|)$. The experiment we ran 20 times and their expectation and standard error have been shown above.

5 CONCLUSION

We have introduced a new MUBs sampler for stochastic trace estimation which combines the efficiency of fixed basis methods with performance which outperforms the state of the art methods. We offer both empirical and theoretical comparisons to the previously established state of the art techniques and clearly demonstrate the benefit of using mutually unbiased bases for stochastic linear algebraic procedures to accelerate machine learning algorithms.

Acknowledgements

JFF acknowledges support from the Singapore Ministry of Education and the US Air Force Office of Scientific Research under AOARD grant FA2386-15-1-4082. This material is based on research funded in part by the Singapore National Research Foundation under NRF Award NRF-NRFF2013-01.

References

[1] Ali Rahimi and Benjamin Recht. Random features for large-scale kernel machines. In *Advances in neural information processing systems*, pages 1177–1184, 2008.

[2] Zhaojun Bai, Mark Fahey, Gene H Golub, M Menon, and E Richter. Computing partial eigenvalue sums in electronic structure calculations. Technical report, Citeseer, 1998.

[3] Tristan van Leeuwen, Aleksandr Y Aravkin, and Felix J Herrmann. Seismic waveform inversion by stochastic optimization. *International Journal of Geophysics*, 2011, 2011.

[4] Eldad Haber, Matthias Chung, and Felix Herrmann. An effective method for parameter estimation with pde constraints with multiple right-hand sides. *SIAM Journal on Optimization*, 22(3):739–757, 2012.

[5] Christos Boutsidis, Petros Drineas, Prabhanjan Kambadur, and Anastasios Zouzias. A randomized algorithm for approximating the log determinant of a symmetric positive definite matrix. *arXiv preprint arXiv:1503.00374*, 2015.

[6] Michael F Hutchinson. A stochastic estimator of the trace of the influence matrix for laplacian smoothing splines. *Communications in Statistics-Simulation and Computation*, 19(2):433–450, 1990.

[7] Mikhail J Atallah, Frédéric Chyzak, and Philippe Dumas. A randomized algorithm for approximate string matching. *Algorithmica*, 29(3):468–486, 2001.

[8] Mikhail J Atallah, Elena Grigorescu, and Yi Wu. A lower-variance randomized algorithm for approxi-

mate string matching. *Information Processing Letters*, 113(18):690–692, 2013.

[9] 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.

[10] Charalampos E Tsourakakis. Fast counting of triangles in large real networks without counting: Algorithms and laws. In *Data Mining, 2008. ICDM'08. Eighth IEEE International Conference on*, pages 608–617. IEEE, 2008.

[11] Michael L Stein, Jie Chen, Mihai Anitescu, et al. Stochastic approximation of score functions for gaussian processes. *The Annals of Applied Statistics*, 7(2):1162–1191, 2013.

[12] Insu Han, Dmitry Malioutov, and Jinwoo Shin. Large-scale log-determinant computation through stochastic chebyshev expansions. In *International Conference on Machine Learning*, pages 908–917, 2015.

[13] Haim Avron and Sivan Toledo. Randomized algorithms for estimating the trace of an implicit symmetric positive semi-definite matrix. *Journal of the ACM (JACM)*, 58(2):8, 2011.

[14] Julian Schwinger. Unitary operator bases. *Proceedings of the National Academy of Sciences*, 46(4):570–579, 1960.

[15] Thomas Durt, Berthold-Georg Englert, Ingemar Bengtsson, and Karol Życzkowski. On mutually unbiased bases. *International journal of quantum information*, 8(04):535–640, 2010.

[16] Michael A Nielsen and Isaac L Chuang. *Quantum computation and quantum information*. Cambridge university press, 2010.

[17] P Oscar Boykin, Meera Sitharam, Mohamad Tarifi, and Pawel Wocjan. Real mutually unbiased bases. *arXiv preprint quant-ph/0502024*, 2005.

[18] Andreas Klappenecker and Martin Rötteler. Constructions of mutually unbiased bases. In *Finite fields and applications*, pages 137–144. Springer, 2004.

[19] Paul Butterley and William Hall. Numerical evidence for the maximum number of mutually unbiased bases in dimension six. *Physics Letters A*, 369(1):5–8, 2007.

[20] Somshubhro Bandyopadhyay, P Oscar Boykin, Vwani Roychowdhury, and Farrokh Vatan. A new proof for the existence of mutually unbiased bases. *Algorithmica*, 34(4):512–528, 2002.

[21] MF Hutchinson. A stochastic estimator of the trace of the influence matrix for laplacian smoothing splines. *Communications in Statistics-Simulation and Computation*, 18(3):1059–1076, 1989.

[22] RN Silver and H Röder. Calculation of densities of states and spectral functions by chebyshev recursion and maximum entropy. *Physical Review E*, 56(4):4822, 1997.

[23] Miller McPherson, Lynn Smith-Lovin, and James M Cook. Birds of a feather: Homophily in social networks. *Annual review of sociology*, 27(1):415–444, 2001.

[24] Stanley Wasserman and Katherine Faust. *Social network analysis: Methods and applications*, volume 8. Cambridge university press, 1994.

[25] Luca Becchetti, Paolo Boldi, Carlos Castillo, and Aristides Gionis. Efficient semi-streaming algorithms for local triangle counting in massive graphs. In *Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 16–24. ACM, 2008.

[26] Jean-Pierre Eckmann and Elisha Moses. Curvature of co-links uncovers hidden thematic layers in the world wide web. *Proceedings of the national academy of sciences*, 99(9):5825–5829, 2002.

[27] Jack Fitzsimons, Kurt Cutajar, Michael Osborne, Stephen Roberts, and Maurizio Filippone. Bayesian inference of log determinants. *arXiv preprint arXiv:1704.01445*, 2017.

[28] Jack Fitzsimons, Diego Granziol, Kurt Cutajar, Michael Osborne, Maurizio Filippone, and Stephen Roberts. Entropic trace estimates for log determinants. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 323–338. Springer, 2017.

[29] R Kelley Pace and James P LeSage. Chebyshev approximation of log-determinants of spatial weight matrices. *Computational Statistics & Data Analysis*, 45(2):179–196, 2004.