POLYNOMIAL TIME ALGORITHMS TO APPROXIMATE PERMANENTS AND MIXED DISCRIMINANTS WITHIN A SIMPLY EXPONENTIAL FACTOR

Alexander Barvinok

Department of Mathematics, University of Michigan
Ann Arbor, MI 48109–1109

Abstract. We present real, complex, and quaternionic versions of a simple randomized polynomial time algorithm to approximate the permanent of a non-negative matrix and, more generally, the mixed discriminant of positive semidefinite matrices. The algorithm provides an unbiased estimator, which, with high probability, approximates the true value within a factor of $O(c^n)$, where n is the size of the matrix (matrices) and where $c \approx 0.28$ for the real version, $c \approx 0.56$ for the complex version and $c \approx 0.76$ for the quaternionic version. We discuss possible extensions of our method as well as applications of mixed discriminants to problems of combinatorial counting. *Random Structures & Algorithms, to appear.*

1. Introduction

In this paper, we construct a family of randomized polynomial time algorithms to approximate the permanent of a non-negative matrix. In particular, one of our algorithms (the quaternionic algorithm of Section 2.3) provides the best known polynomial time approximation for the permanent of an arbitrary non-negative matrix. Our approximation algorithms generalize naturally to mixed discriminants, quantities of independent interest. Possible extensions of our method and applications of mixed discriminants to problems of combinatorial counting are discussed in the last two sections.

(1.1) Permanent. Let $A = (a_{ij})$ be an $n \times n$ matrix and let S_n be the symmetric group, that is the group of all permutations of the set $\{1, \ldots, n\}$. The number

$$\mathrm{per}\ A = \sum_{\sigma \in S_n} \prod_{i=1}^{n} a_{i\sigma(i)}$$

Key words and phrases. permanent, mixed discriminant, randomized algorithms, approximation algorithms.

This research was partially supported by Alfred P. Sloan Research Fellowship, by NSF grants DMS 9501129 and DMS 9734138, and by the Mathematical Sciences Research Institute, Berkeley, CA through NSF grant DMS 9022140.

Typeset by $\mathcal{A}\mathcal{M}\mathcal{S}$-TEX

is called the *permanent* of A. We assume that A is non-negative, that is $a_{ij} \geq 0$ for all $i, j = 1, \ldots, n$. If A is a 0-1 matrix, then per A can be interpreted as the number of perfect matchings in a bipartite graph G on $2n$ vertices $v_1, \ldots, v_n$ and $u_1, \ldots, u_n$, where (v_i, u_j) is an edge of G if and only if $a_{ij} = 1$. To compute the permanent of a given 0-1 matrix is a #P-complete problem and even to estimate per A seems to be difficult. Polynomial time algorithms for computing per A are known when A has some special structure, for example, when A has a small rank [5], or when A is a 0-1 matrix and per A is small (see [14] and Section 7.3 of [25]).

Since the exact computation is difficult, a natural question is how well one can approximate the permanent in polynomial time. In particular, is it true that for any $\epsilon > 0$ there is a polynomial time (possibly randomized) algorithm that approximates the permanent of a given matrix within a relative error ϵ? In other words, does there exist a polynomial time approximation scheme? Polynomial time approximation schemes are known for dense 0-1 matrices [15], for "almost all" 0-1 matrices (see [15], [12], and [27]) and for some special 0-1 matrices, such as those corresponding to lattice graphs (see [16] for a survey on approximation algorithms). However, no polynomial time approximation scheme is known for an arbitrary 0-1 matrix (see [18] for the fastest known "mildly exponential" approximation scheme).

In [6], the author suggested a polynomial time randomized algorithm, which, given an $n \times n$ non-negative matrix A, outputs a non-negative number α approximating per A within a simply exponential in n factor. The algorithm uses randomization, so α is a random variable. The expectation of α is per A and with high probability (say, with probability at least 0.9) we have

$$c^n \operatorname{per} A \leq \alpha \leq C \operatorname{per} A, \tag{1.1.1}$$

where C and $c > 0$ are absolute constants (with $c \approx 0.28$). As usual, the probability 0.9 can be improved to $1 - \epsilon$ by running the algorithm independently $O(\log \epsilon^{-1})$ times and choosing α to be median of the computed α's.

Recently, N. Linial, A. Samorodnitsky, and A. Wigderson [22] constructed a polynomial time deterministic algorithm, which achieves (1.1.1) with $C = 1$ and $c = 1/e \approx 0.37$. The algorithm uses scaling of a given non-negative matrix to a doubly stochastic matrix.

In this paper, we present a family of randomized polynomial time algorithms for approximating the permanent within a simply exponential factor. We present real, complex, and quaternionic versions of an unbiased estimator, each achieving a better degree of approximation than the previous one. Our estimators produce a number α whose expectation is the permanent and which with high probability satisfies (1.1.1), where $c \approx 0.28$ for the real algorithm, $c \approx 0.56$ for the complex algorithm, and $c \approx 0.76$ for the quaternionic algorithm. The last algorithm provides the best known polynomial time approximation for the permanent of an arbitrary non-negative matrix. The algorithms have a much simpler structure and are easier to implement than the algorithm of [6].

The first version (see [7]) of the paper contained the real algorithm only. The complex algorithm was suggested to the author by M. Dyer and M. Jerrum [8]. Building on the complex version, the author constructed the quaternionic version.

(1.2) Mixed Discriminant. Let $Q_1, \ldots, Q_n$ be $n \times n$ real symmetric matrices and let $t_1, \ldots, t_n$ be variables. Then $\det(t_1 Q_1 + \ldots + t_n Q_n)$ is a homogeneous polynomial of degree n in $t_1, \ldots, t_n$. The number

$$D(Q_1, \ldots, Q_n) = \frac{\partial^n}{\partial t_1 \ldots \partial t_n} \det\bigl(t_1 Q_1 + \ldots + t_n Q_n\bigr)$$

is called the mixed discriminant of $Q_1, \ldots, Q_n$. Sometimes the normalizing factor $1/n!$ is used (cf. [21]). The mixed discriminant $D(Q_1, \ldots, Q_n)$ is a polynomial in the entries of $Q_1, \ldots, Q_n$: for $Q_k = (q_{ij,k})$: $i, j = 1, \ldots, n$, $k = 1, \ldots, n$, we have

$$D(Q_1, \ldots, Q_n) = \sum_{\sigma_1, \sigma_2 \in S_n} (\operatorname{sgn} \sigma_1)(\operatorname{sgn} \sigma_2) \prod_{k=1}^{n} q_{\sigma_1(k)\sigma_2(k),k}. \tag{1.2.1}$$

The mixed discriminant can be considered as a generalization of the permanent. Indeed, from (1.2.1) we deduce that for diagonal matrices $Q_1, \ldots, Q_n$, where $Q_i = \operatorname{diag}\{a_{i1}, \ldots, a_{in}\}$, we have

$$D(Q_1, \ldots, Q_n) = \operatorname{per}\ A, \quad \text{where} \quad A = (a_{ij}).$$

Mixed discriminants were introduced by A.D. Aleksandrov in his proof of the Aleksandrov - Fenchel inequality for mixed volumes ([2], see also [21]). The relation between the mixed discriminant and the permanent was used in the proof of the van der Waerden conjecture for permanents of doubly stochastic matrices (see [9]).

It is known that $D(Q_1, \ldots, Q_n) \geq 0$ provided $Q_1, \ldots, Q_n$ are positive semidefinite (see [21]). Just as it is natural to restrict the permanent to non-negative matrices, it is natural to restrict the mixed discriminant to positive semidefinite matrices.

Mixed discriminants generalize permanents but they also have some independent applications to computationally hard problems of combinatorial counting, some of which we describe in this paper. Suppose, for example, we are given a connected graph with $n + 1$ vertices, whose edges are colored in n colors. Then the number of spanning trees having exactly one edge of each color can be expressed as the mixed discriminant of some n positive semidefinite matrices, explicitly computed from the incidence matrix of the graph. Mixed discriminants play an important role in convex and integral geometry (see [21]) and the problem of their efficient computation/approximation is not less interesting (but certainly less publicized) than the problem of efficient computation/approximation of permanents.

In [6], the author suggested a randomized polynomial time algorithm, which, given n positive semidefinite matrices $Q_1, \ldots, Q_n$, computes a number α, which with high probability satisfies the inequalities

$$c^n D(Q_1, \ldots, Q_n) \leq \alpha \leq C \cdot D(Q_1, \ldots, Q_n),$$

where $c > 0$ and C are absolute constants (with $c \approx 0.28$). In this paper, we construct a family of algorithms (again, real, complex, and quaternionic), which achieve $c \approx 0.28$ (real), $c \approx 0.56$ (complex), and $c \approx 0.76$ (quaternionic). The algorithms are natural generalizations of the permanent approximation algorithms. The real algorithm (Section 3.1) can be interpreted as a "parallelization" of the algorithm from [6]. One can note that the permanent approximation algorithm of [22] does not obviously generalize to mixed discriminants.

The paper is organized as follows.

In Section 2, we describe the permanent approximation algorithms.

In Section 3, we describe the mixed discriminant approximation algorithms.

In Section 4, we prove some general inequalities for quadratic forms on Euclidean spaces endowed with a Gaussian measure. The results of the section constitute the core of our proofs.

In Section 5, we prove a technical martingale-type result, which we use in our proofs.

In Section 6, we prove the main results of the paper.

In Section 7, we discuss possible extensions of our method. In particular, we discuss how to approximate hafnians and sums of subpermanents of a rectangular matrix.

In Section 8, we discuss some applications of mixed discriminants to counting.

(1.3) Notation. Our approximation constants belong to a certain family which we define here. For a positive integer k let $x_1, \ldots, x_k$ be independent real valued random variables having the Gaussian distribution with the density

$$\psi(x) = \sqrt{\frac{k}{2\pi}} \exp\Big\{-\frac{kx^2}{2}\Big\}.$$

Thus the expectation of x_i is 0 and the expectation of x_i^2 is $1/k$, so the expected value of $x_1^2 + \ldots + x_k^2$ is 1. Let us define $\mathfrak{C}_k$ to be the expectation of $\ln(x_1^2 + \ldots + x_k^2)$; that is,

$$\mathfrak{C}_k = \Big(\frac{k}{2\pi}\Big)^{k/2} \int_{\mathbb{R}^k} \ln(x_1^2 + \ldots + x_k^2) \exp\Big\{-\frac{k(x_1^2 + \ldots + x_k^2)}{2}\Big\}\, dx_1 \ldots dx_k.$$

In particular, we will be interested in the following values:

$$\mathfrak{C}_1 \approx -1.270362845, \quad \mathfrak{C}_2 \approx -0.5772156649, \quad \text{and} \quad \mathfrak{C}_4 \approx -0.2703628455.$$

One can show that $\mathfrak{C}_1 = -\gamma - \ln 2$, $\mathfrak{C}_2 = -\gamma$, and $\mathfrak{C}_4 = 1 - \gamma - \ln 2$, where $\gamma \approx 0.5772156649$ is the Euler constant. Let us define

$$\mathfrak{c}_k = \exp\{\mathfrak{C}_k\}.$$

In particular,

$$\mathfrak{c}_1 = \frac{e^{-\gamma}}{2} \approx 0.2807297419, \quad \mathfrak{c}_2 = e^{-\gamma} \approx 0.5614594836, \quad \text{and}$$

$$\mathfrak{c}_4 = \frac{e^{1-\gamma}}{2} \approx 0.7631025558.$$

It turns out that $\mathfrak{c}_1$ is the approximation constant in the real algorithms, $\mathfrak{c}_2$ is the approximation constant in the complex algorithms, and $\mathfrak{c}_4$ is the approximation constant in the quaternionic algorithms.

One can argue that the quaternionic version of our algorithms is more efficient than the complex version and that the latter is more efficient than the real version. However, the author believes that all three versions are of interest, and, consequently, discusses all three in some detail. The general method can be extended to other problems of approximate counting (see Section 7) and it may happen, for example, that there is an obvious real extension, whereas the existence of a quaternionic extension is problematic.

2. Permanent Approximation Algorithms

In this section, we present the three versions of the permanent approximation algorithm: real, complex, and quaternionic. The algorithms are very similar: their input is an $n \times n$ non-negative matrix $A = (a_{ij})$ and the output is a non-negative number α, approximating per A. Our algorithms are randomized. In the course of the computation, we generate a random variable, so the output α is a random variable. Specifically, we use sampling from the Gaussian distribution in $\mathbb{R}^1$ with the density $\psi_\sigma(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{x^2}{2\sigma^2}}$. As is known (see, for example, Section 3.4.1 of [20]), sampling from the Gaussian distribution can be efficiently simulated from the standard Bernoulli distribution (sampling a random bit). The output α is an unbiased estimator for per A and it turns out that α is unlikely to overestimate per A by a large constant factor (easy to prove) and unlikely to underestimate per A by a factor of $O(c^n)$, where $c > 0$ is an absolute constant (harder to prove). For example, it follows that for all sufficiently large n, the probability that α satisfies the inequalities

$$c^n \operatorname{per} A \leq \alpha \leq 3 \operatorname{per} A$$

is at least 0.6. As we noted in Section 1, for any $\epsilon > 0$ we can improve the probability to $1 - \epsilon$ by running the algorithm $O(\log \epsilon^{-1})$ times and choosing the median of the computed α's, cf. [17]. We can choose $c = 0.28$ in the real algorithm, $c = 0.56$ in the complex algorithm, and $c = 0.76$ in the quaternionic algorithm.

The computational model is the RAM (Random Access Machine) with the uniform cost criterion (see [1]), so the algorithms operate with real numbers and allow arithmetic operations (addition, subtraction, multiplication, and division) and comparison of real numbers. A preprocessing requires taking the square root of each entry of the matrix A.

The computational complexity of the algorithms is bounded by a polynomial in n. Each subsequent algorithm attains a better degree of approximation but is more time-consuming. The real version reduces to the computation of the determinant of an $n \times n$ real matrix, the complex version reduces to the computation of the determinant of an $n \times n$ complex matrix, and the the quaternionic version reduces to the computation of the determinant of a $2n \times 2n$ complex matrix. As is known, the determinant of an $n \times n$ real or complex matrix can be routinely computed using $O(n^3)$ arithmetic operations. Our algorithms can be rewritten in the binary ("Turing") mode, so that they remain polynomial time. This transformation is straightforward; it is sketched in the first version of the paper [7]. The general technique of going from "real randomized" to "binary randomized" algorithms can be found in [24].

(2.1) The Real Algorithm.

Input: A non-negative $n \times n$ matrix A.

Output: A non-negative number α approximating $\operatorname{per} A$.

Algorithm: Sample independently n^2 numbers $u_{ij} : i, j = 1, \ldots, n$ at random from the standard Gaussian distribution in $\mathbb{R}$ with the density $\psi_{\mathbb{R}}(x) = \dfrac{1}{\sqrt{2\pi}} e^{-x^2/2}$. Compute the $n \times n$ matrix $B = (b_{ij})$, where $b_{ij} = u_{ij}\sqrt{a_{ij}}$. Compute $\alpha = (\det B)^2$. Output α.

(2.1.1) Theorem.

(1) *The expectation of α is* $\operatorname{per} A$.

(2) *For any $C > 1$ the probability that*

$$\alpha \geq C \cdot \operatorname{per} A$$

does not exceed C^{-1}.

(3) *For any $1 > \epsilon > 0$ the probability that*

$$\alpha \leq (\epsilon \mathfrak{c}_1)^n \operatorname{per} A$$

does not exceed $8/(n \ln^2 \epsilon)$, where $\mathfrak{c}_1 \approx 0.28$ is the constant defined in Section 1.3.

Remark. Relation to the Godsil-Gutman estimator. It is immediately seen that Algorithm 2.1 is a modification of the Godsil-Gutman estimator (see [13]) and Chapter 8 of [23]). Indeed, in the Godsil-Gutman estimator we sample u_{ij} from the binary distribution:

$$u_{ij} = \begin{cases} 1 & \text{with probability } 1/2 \\ -1 & \text{with probability } 1/2. \end{cases}$$

Furthermore, Part 1 and Part 2 of Theorem 2.1.1 remain true as long as we sample u_{ij} independently from some distribution with the expectation 0 and variance 1.

However, Part 3 does not hold true for the binary distribution. In fact, one can show that as long as u_{ij} are sampled from some fixed *discrete* distribution (or, more generally, from some distribution with atoms), there exist 0-1 matrices with arbitrarily large permanents, for which the expected number of trials to produce a non-zero α is exponentially large. Indeed, consider a version of the Godsil-Gutman estimator, where u_{ij} can accept a certain value with a positive probability. Let us fix an integer $k > 1$ and let us consider the $k \times k$ matrix J_k filled with 1's. If we multiply the (i, j)-th entry of J_k by u_{ij}, there is some positive probability $p > 0$ that we get a matrix with two identical rows, so the obtained matrix will have the zero determinant. For $n > 0$, consider an $(nk) \times (nk)$ matrix A which consists of n diagonal blocks J_k. Then $\text{per } A_n = (\text{per } J_k)^n = (k!)^n$. However, the probability that the output $\alpha = 0$ is at least $1 - (1 - p)^n$, which approaches 1 exponentially fast as n grows. Note that even the scaled value $(\text{per } A_n)^{1/nk}$ can be made arbitrarily large, since $(\text{per } A_n)^{1/nk} = (k!)^{1/k} \approx k/e$ for large k. For example, let us choose $k = 8$. Then A_n is a 0-1 $8n \times 8n$ matrix and $\text{per } A_n = (40,320)^n$. The described above binary version of the Godsil-Gutman estimator outputs 0 with the probability at least $1-(0.8)^n$. The complex discrete version of [19] (see also Remark in Section 2.2) outputs 0 with the probability at least $1 - (0.99)^n$. Algorithm 2.1 outputs at least $(1.52)^n$ with the probability approaching 1 as $n \longrightarrow +\infty$. Similarly, the complex version (Algorithm 2.2) outputs at least $(390)^n$ and the quaternionic version (Algorithm 2.3) outputs at least $(4,487)^n$ with the probability approaching 1 as $n \longrightarrow +\infty$.

Of course, for other matrices (for example, for the identity matrix) and even for the majority of matrices (cf. [12]) the binary version of the estimator may perform better than Algorithm 2.1. However, any version of the Godsil-Gutman estimator which approximates the permanent of an *arbitrary* 0-1 matrix within *some* positive constant in expected polynomial time must either use a continuous distribution or a sequence of discrete distributions which depend on the size of the matrix.

(2.2) The Complex Algorithm. By the standard complex Gaussian distribution in $\mathbb{C}$ we mean the distribution of a complex random variable $z = x + iy$ with the density

$$\psi_{\mathbb{C}}(z) = \frac{1}{\pi} e^{-(x^2+y^2)} = \frac{1}{\pi} e^{-|z|^2}.$$

Here $|z| = \sqrt{x^2 + y^2}$ is the absolute value of z. Note that the expectation of $|z|^2$ is 1. To sample from this distribution, it suffices to sample the real and imaginary parts independently from the Gaussian distribution in $\mathbb{R}^1$ with the density $\dfrac{1}{\sqrt{\pi}} e^{-x^2}$.

Our algorithm is the following:

Input: A non-negative $n \times n$ matrix A.

Output: A non-negative number α approximating $\text{per } A$.

Algorithm: Sample independently n^2 numbers u_{ij} at random from the standard complex Gaussian distribution in $\mathbb{C}$ with the density $\psi_{\mathbb{C}}$. Compute the $n \times n$ matrix $B = (b_{ij})$, where $b_{ij} = u_{ij}\sqrt{a_{ij}}$. Compute $\alpha = |\det B|^2$.
Output α.

(2.2.1) Theorem.

(1) *The expectation of* α *is* per A.

(2) *For any $C > 1$ the probability that*

$$\alpha \geq C \cdot \operatorname{per} \ A$$

does not exceed C^{-1}.

(3) *For any $1 > \epsilon > 0$ the probability that*

$$\alpha \leq (\epsilon \mathfrak{c}_2)^n \operatorname{per} \ A$$

does not exceed $8/(n \ln^2 \epsilon)$, where $\mathfrak{c}_2 \approx 0.56$ is the constant defined in Section 1.3.

Remark. Relation to the Karmarkar-Karp-Lipton-Lovász-Luby estimator.
It is seen that Algorithm 2.2 is a simple modification of the estimator from [19]. In [19], the authors sample u_{ij} from the set of roots of unity of degree 3 and then use averaging over a large number of trials. The goal of [19] is somewhat dual to our goal. We want to minimize the error of approximation keeping the running time polynomial whereas the authors of [19] want to minimize the time needed to approximate the permanent keeping the relative error of approximation at most $(1+\epsilon)$, where $\epsilon > 0$ is a part of the input. The complexity of the algorithm from [19] is $poly(n)\epsilon^{-2}2^{n/2}$ and, while still exponential, is better than $n2^n$ complexity of Ryser's exact algorithm (see Section 7.2 of [25]). Obviously, both approaches have their advantages. Algorithms 2.1–2.3 provide a quick rough estimate, whereas the algorithms from [18] and [19] are much more precise but also more time-consuming. It would be interesting to find out whether by applying repeated sampling and averaging in Algorithm 2.2, one can get the running time to achieve the relative error ϵ comparable to that of [19], and if that can be extended to arbitrary non-negative matrices (as the performance of the algorithm from [19] is guaranteed for 0-1 matrices only).

(2.3) The Quaternionic Algorithm. We recall that the algebra $\mathbb{H}$ of quaternions is the 4-dimensional real vector space with the basis vectors $1, \mathrm{i}, \mathrm{j}, \mathrm{k}$ that satisfy the following multiplication rules:

$$\mathrm{i}^2 = \mathrm{j}^2 = \mathrm{k}^2 = -1,$$

$$\mathrm{i}\,\mathrm{j} = -\,\mathrm{j}\,\mathrm{i} = \mathrm{k}, \quad \mathrm{j}\,\mathrm{k} = -\,\mathrm{k}\,\mathrm{j} = \mathrm{i}, \quad \mathrm{k}\,\mathrm{i} = -\,\mathrm{i}\,\mathrm{k} = \mathrm{j},$$

and

$$1\,\mathrm{i} = \mathrm{i}\,1 = \mathrm{i}, \quad 1\,\mathrm{j} = \mathrm{j}\,1 = \mathrm{j}, \quad 1\,\mathrm{k} = \mathrm{k}\,1 = \mathrm{k}.$$

The norm $|h|$ of a quaternion $h = a + \mathrm{i}\,b + \mathrm{j}\,c + \mathrm{k}\,d$, where $a, b, c, d \in \mathbb{R}$, is defined as $|h| = \sqrt{a^2+b^2+c^2+d^2}$. By the standard quaternionic Gaussian distribution in $\mathbb{H}$ we mean the distribution of a quaternionic random variable $h = a + \mathrm{i}b + \mathrm{j}c + \mathrm{k}d$ with the density

$$\psi_{\mathbb{H}}(h) = \frac{4}{\pi^2} e^{-2(a^2+b^2+c^2+d^2)}.$$

In particular, the expectation of $|h|^2$ is 1. To sample from the distribution, it suffices to sample a, b, c, and d independently from the Gaussian distribution in $\mathbb{R}^1$ with the density $\sqrt{\frac{2}{\pi}}e^{-2x^2}$.

Our algorithm is the following:

Input: A non-negative $n \times n$ matrix A.

Output: A number α approximating per A.

Algorithm: Sample independently n^2 quaternions u_{ij} at random from the standard quaternionic Gaussian distribution in $\mathbb{H}$ with the density $\psi_{\mathbb{H}}$. Compute the $n \times n$ quaternionic matrix $H = (h_{ij})$, where $h_{ij} = u_{ij}\sqrt{a_{ij}}$. Write $H = R + \mathrm{i}\,B + \mathrm{j}\,C + \mathrm{k}\,D$, where R, B, C, and D are real $n \times n$ matrices. Construct the $2n \times 2n$ complex matrix

$$H_{\mathbb{C}} = \begin{pmatrix} R + iB & C + iD \\ -C + iD & R - iB \end{pmatrix}.$$

Compute $\alpha = \det H_{\mathbb{C}}$.
Output α.

(2.3.1) Theorem.

(1) *The output α is a non-negative real number and its expectation is* per A.

(2) *For any $C > 1$ the probability that*

$$\alpha \geq C \cdot \text{per } A$$

does not exceed C^{-1}.

(3) *For any $1 > \epsilon > 0$ the probability that*

$$\alpha \leq (\epsilon \mathfrak{c}_4)^n \text{ per } A$$

does not exceed $8/(n \ln^2 \epsilon)$, where $\mathfrak{c}_4 \approx 0.76$ is the constant defined in Section 1.3.

Remark. Relations to the real and complex estimators. An obvious obstacle to constructing a quaternionic version of Algorithms 2.1 and 2.2 is that it appears that we need to use the "determinant" of a quaternionic matrix. However, a closer look reveals that what we need is *the squared norm* of the determinant. This "squared norm of the determinant" can be defined using the canonical 2-dimensional complex representation of the quaternions. It turns out that that the determinant of the complex matrix $H_{\mathbb{C}}$, computed in Algorithm 2.3, is the right definition of the "squared norm of the determinant" of a quaternionic matrix H (it is known as the "reduced norm", or the squared norm of the Dieudonné determinant, see Chapter VI, Section 1 of [3]). As an analogue, let us point out that the the squared absolute value of the determinant of an $n \times n$ complex matrix can be interpreted as the determinant of a $2n \times 2n$ real matrix:

$$|\det(A + iB)|^2 = \det \begin{pmatrix} A & B \\ -B & A \end{pmatrix}.$$

Example. The identity matrix. The following example provides some intuition why the quaternionic estimator gives a better approximation than the complex estimator and why the complex estimator gives a better approximation than the real estimator. Suppose that $A = I$ is the $n \times n$ identity matrix, so per $A = 1$. Algorithm 2.1 approximates 1 by the product

$$\alpha = x_1^2 \cdots x_n^2,$$

where $x_i = u_{ii}$ are independent random variables sampled from the standard Gaussian distribution in $\mathbb{R}^1$. The expectation $\mathbf{E}(x_i^2)$ of each factor is 1, so the expected value of the product is 1. However, since the values of x_i^2 can deviate from the expectation, we can accumulate an exponentially large deviation. Since

$$\frac{\ln \alpha}{n} = \frac{1}{n} \sum_{i=1}^{n} \ln x_i^2,$$

the law of large numbers implies that $(\ln \alpha)/n$ concentrates around $\mathfrak{C}_1 = \mathbf{E}(\ln x_i^2)$ (cf. Section 1.3), so α is reasonably close to $\exp\{n\mathfrak{C}_1\} = \mathfrak{c}_1^n$ most of the time.

Algorithm 2.2 approximates 1 by the product

$$\alpha = \left(\frac{x_1^2 + y_1^2}{2}\right) \cdots \left(\frac{x_n^2 + y_n^2}{2}\right),$$

where x_i, y_i are independent random variables sampled from the standard Gaussian distribution in $\mathbb{R}^1$. Again, the expectation of each factor is 1, but because of the averaging, $\dfrac{x_i^2 + y_i^2}{2}$ is concentrated around its expectation somewhat more sharp than either x_i^2 or y_i^2. Therefore, the accumulated error, while still exponential, is smaller than that in Algorithm 2.1. Indeed, we have (cf. Section 1.3)

$$\mathfrak{c}_2 = \exp\left\{\mathbf{E}\left(\ln \frac{x_i^2 + y_i^2}{2}\right)\right\} > \mathfrak{c}_1.$$

Finally, Algorithm 2.3 approximates 1 by the product

$$\alpha = \left(\frac{a_1^2 + b_1^2 + c_1^2 + d_1^2}{4}\right) \cdots \left(\frac{a_n^2 + b_n^2 + c_n^2 + d_n^2}{4}\right),$$

where a_i, b_i, c_i, d_i are independent random variables sampled from the standard Gaussian distribution in $\mathbb{R}^1$. Here we have a still sharper concentration of each factor around its expectation, so the total error gets smaller. For the constant $\mathfrak{c}_4$ we have (cf. Section 1.3)

$$\mathfrak{c}_4 = \exp\left\{\mathbf{E}\left(\ln \frac{a_i^2 + b_i^2 + c_i^2 + d_i^2}{4}\right)\right\} > \mathfrak{c}_2.$$

It appears that the Algorithms 2.1–2.3 are related to Clifford algebras $\mathbb{R}$, $\mathbb{C}$, and $\mathbb{H}$ with 0, 1, and 3 generators respectively (cf. Section 41 of [26]). There seem to be ways to associate an approximation algorithm with any Clifford algebra, but it is not clear at the moment whether those algorithms can be of any interest. It would be interesting to find out whether for any k there is a version of our algorithm achieving a $\mathfrak{c}_k^n$ approximation (we note that $\mathfrak{c}_k \longrightarrow 1$ as $k \longrightarrow +\infty$).

Remark. How well can we approximate the permanent in polynomial time ? Suppose we have a polynomial time (probabilistic or deterministic) algorithm that for any given $n \times n$ (non-negative or 0-1) matrix A computes a number α such that

$$\phi(n) \operatorname{per} \ A \leq \alpha \leq \operatorname{per} \ A.$$

What sort of function might $\phi(n)$ be? For an $n \times n$ matrix A and $k > 0$, let us construct the $nk \times nk$ block-diagonal matrix A_k, having k diagonal copies of A. We observe that A_k is non-negative if A is non-negative and A_k is 0-1 if A is 0-1, and that $\operatorname{per} \ A = \big(\operatorname{per} \ A_k\big)^{1/k}$. Applying our algorithm to A_k and taking the root we get an approximation α_k, where

$$(\phi(nk))^{1/k} \operatorname{per} \ A \leq \alpha_k \leq \operatorname{per} \ A,$$

Therefore, we can always improve ϕ to $\phi_k = (\phi(nk))^{1/k}$, where k must be bounded by a polynomial in n to keep polynomial time complexity. For example, if $\phi(n) = n^{-\beta}$ for some $\beta > 0$ or if $\phi(n) = \exp\{-n^{\beta}\}$ for some $0 < \beta < 1$, then by choosing a sufficiently large $k = k(n)$ we can always improve ϕ to $\phi_k = (1 - \epsilon)$ with any given $1 > \epsilon > 0$.

There are few obvious choices for "non-improvable" functions ϕ.

a) $\phi(n) \equiv 1$. This doesn't look likely, given that the problem is #P-hard.

b) For any $\epsilon > 0$ one can choose $\phi_\epsilon(n) = 1 - \epsilon$, and the algorithm is polynomial in ϵ^{-1}. In the author's opinion this conjecture is overly optimistic.

c) For any $\epsilon > 0$ one can choose $\phi_\epsilon(n) = (1 - \epsilon)^n$, but the algorithm is *not* polynomial in ϵ^{-1}. The existence of this type of approximation was conjectured by V.D. Milman.

d) $\phi(n) = c^n$ for some fixed constant c. This is the type of a bound achieved by Algorithms 2.1–2.3. An interesting question is, what is the best possible constant c?

3. Mixed Discriminant Approximation Algorithms

It turns out that the permanent approximation algorithms of Section 2 can be naturally extended to mixed discriminants. The input of the algorithms consists of $n \times n$ positive semidefinite matrices $Q_1, \ldots, Q_n$ and the output is a non-negative number α approximating $D(Q_1, \ldots, Q_n)$. As in Section 2, we use the real model of computation. A preprocessing requires representing each matrix Q_i as the product $Q_i = T_i T_i^*$ of a real matrix and its transpose. This is a standard procedure of linear algebra, which requires $O(n^3)$ arithmetic operations and n square root extractions (see, for example, Chapter 2, Section 10 of [11]).

(3.1) The Real Algorithm. The algorithm requires sampling vectors $x \in \mathbb{R}^n$, $x = (\xi_1, \ldots, \xi_n)$ from the standard Gaussian distribution in $\mathbb{R}^n$ with the density

$$\psi_{\mathbb{R}^n}(x) = (2\pi)^{-n/2} \exp\{-\|x\|^2/2\}, \quad \text{where} \quad \|x\|^2 = \xi_1^2 + \ldots + \xi_n^2.$$

We interpret x as an n-column of real numbers $\xi_1, \ldots, \xi_n$. Note that the expectation of ξ_i^2 is 1 and that the expectation of $\ln \xi_i^2$ is $\mathfrak{C}_1$ (see Section 1.3). To sample from the distribution, we can sample the coordinates $\xi_1, \ldots, \xi_n$ independently from the one-dimensional standard Gaussian distribution with the density $\psi_{\mathbb{R}} = \dfrac{1}{\sqrt{2\pi}} e^{-\frac{x^2}{2}}$, cf. Section 2.1. Our algorithm is the following:

Input: Positive semidefinite $n \times n$ matrices $Q_1, \ldots, Q_n$.
Output: A number α approximating the mixed discriminant $D(Q_1, \ldots, Q_n)$.
Algorithm: For $i = 1, \ldots, n$ compute a decomposition $Q_i = T_i T_i^*$. Sample independently n vectors $u_1, \ldots, u_n$ at random from the standard Gaussian distribution in $\mathbb{R}^n$ with the density $\psi_{\mathbb{R}^n}(x)$. Compute

$$\alpha = \Big(\det[T_1 u_1, \ldots, T_n u_n]\Big)^2,$$

the squared determinant of the matrix with the columns $T_1 u_1, \ldots, T_n u_n$.
Output α.

(3.1.1) Theorem.

(1) *The expectation of α is the mixed discriminant $D(Q_1, \ldots, Q_n)$.*
(2) *For any $C > 1$ the probability that*

$$\alpha \geq C \cdot D(Q_1, \ldots, Q_n)$$

does not exceed C^{-1}.
(3) *For any $1 > \epsilon > 0$ the probability that*

$$\alpha \leq (\epsilon \mathfrak{c}_1)^n D(Q_1, \ldots, Q_n)$$

does not exceed $8/(n \ln^2 \epsilon)$, where $\mathfrak{c}_1 \approx 0.28$ is the constant defined in Section 1.3.

Remark. Relation to the estimator from [6]. The first randomized polynomial time algorithm to approximate the mixed discriminant within an exponential factor was constructed in the author's paper [6]. It achieves asymptotically the same degree of approximation as Algorithm 3.1. The idea of the algorithm from [6] is to apply repeatedly the following two steps: first, by applying an appropriate linear transformation, we reduce the problem to the special case, where $Q_n = I$ is the identity matrix. Next, we replace $D(Q_1, \ldots, Q_{n-1}, I)$ by $nD(Q_1', \ldots, Q_{n-1}')$, where Q_i' are $(n-1) \times (n-1)$ symmetric matrices interpreted as projections of Q_i onto a randomly chosen linear hyperplane in $\mathbb{R}^n$. Algorithm 3.1 can be interpreted as a "parallelization" of the algorithm from [6]. Instead of the successive projections, we independently project the matrices Q_i onto randomly chosen lines in $\mathbb{R}^n$.

(3.2) The Complex Algorithm. The algorithm requires sampling random complex vectors $z = (\zeta_1, \ldots, \zeta_n) \in \mathbb{C}^n$ from the standard Gaussian distribution with the density

$$\psi_{\mathbb{C}^n}(z) = \frac{1}{\pi^n} \exp\{-\|z\|^2\}, \quad \text{where} \quad \|z\|^2 = |\zeta_1|^2 + \ldots + |\zeta_n|^2.$$

We interpret z as an n-column of complex numbers $\zeta_1, \ldots, \zeta_n$. Note that the expectation of $|\zeta_i|^2$ is 1 and that the expectation of $\ln |\zeta_i|^2$ is $\mathfrak{C}_2$ (see Section 1.3). To sample from the distribution, we can sample each ζ_i independently from the standard one-dimensional complex Gaussian distribution with the density $\psi_{\mathbb{C}}(\zeta) = \dfrac{1}{\pi} e^{-|\zeta|^2}$, cf. Section 2.2. Our algorithm is the following:

Input: Real positive semidefinite $n \times n$ matrices $Q_1, \ldots, Q_n$.

Output: A real number α approximating the mixed discriminant $D(Q_1, \ldots, Q_n)$.

Algorithm: For $i = 1, \ldots, n$ compute a decomposition $Q_i = T_i T_i^*$. Sample independently n vectors $u_1, \ldots, u_n$ at random from the standard complex Gaussian distribution in $\mathbb{C}^n$.

Compute

$$\alpha = \Big| \det[T_1 u_1, \ldots, T_n u_n] \Big|^2,$$

the squared absolute value of the determinant of the matrix with the columns $T_1 u_1, \ldots, T_n u_n$.
Output α.

(3.2.1) Theorem.

(1) *The expectation of α is the mixed discriminant $D(Q_1, \ldots, Q_n)$.*

(2) *For any $C > 1$ the probability that*

$$\alpha \geq C \cdot D(Q_1, \ldots, Q_n)$$

does not exceed C^{-1}.

(3) *For any $1 > \epsilon > 0$ the probability that*

$$\alpha \leq (\epsilon \mathfrak{c}_2)^n D(Q_1, \ldots, Q_n)$$

does not exceed $8/(n \ln^2 \epsilon)$, where $\mathfrak{c}_2 \approx 0.56$ is the constant defined in Section 1.3.

(3.3) The Quaternionic Algorithm. Let $\mathbb{H}^n$ be the set of all n-tuples (vectors) $h = (\tau_1, \ldots, \tau_n)$ of quaternions $\tau_i \in \mathbb{H}$. For an $n \times n$ real matrix T and a vector $h = a + \mathrm{i}b + \mathrm{j}c + \mathrm{k}d \in \mathbb{H}^n$, where $a, b, c, d \in \mathbb{R}^n$, by $Th \in \mathbb{H}^n$ we understand the vector $Ta + \mathrm{i}Tb + \mathrm{j}Tc + \mathrm{k}Td$. The algorithm requires sampling a random vector

$h = (\tau_1, \dots, \tau_n)$ from the standard quaternionic Gaussian distribution in $\mathbb{H}^n$ with the density

$$\psi_{\mathbb{H}^n}(h) = \frac{4^n}{\pi^{2n}} \exp\{-2\|h\|^2\}, \quad \text{where} \quad \|h\|^2 = |\tau_1|^2 + \ldots + |\tau_n|^2.$$

We interpret h as an n-column of quaternions $\tau_1, \dots, \tau_n$. Note that the expectation of $|\tau_i|^2$ is 1 and that the expectation of $\ln|\tau_i|^2$ is $\mathfrak{C}_4$ (see Section 1.3). To sample from the distribution, we can sample each $\tau_i \in \mathbb{H}$ independently from the standard one-dimensional quaternionic Gaussian distribution with the density $\psi_{\mathbb{H}}(\tau) = \frac{4}{\pi^2} e^{-2|\tau|^2}$, cf. Section 2.3. Our algorithm is the following:

Input: Real positive semidefinite $n \times n$ matrices $Q_1, \dots, Q_n$.

Output: A number α approximating the mixed discriminant $D(Q_1, \dots, Q_n)$.

Algorithm: For $i = 1, \dots, n$ compute a decomposition $Q_i = T_i T_i^*$. Sample independently n vectors $u_1, \dots, u_n$ at random from the standard quaternionic Gaussian distribution in $\mathbb{H}^n$. Compute the $n \times n$ quaternionic matrix $H = [T_1 u_1, \dots, T_n u_n]$, whose columns are $T_1 u_1, \dots, T_n u_n$. Write $H = A + \mathrm{i}B + \mathrm{j}C + \mathrm{k}D$, where A, B, C, and D are real $n \times n$ matrices. Construct the $2n \times 2n$ complex matrix

$$H_{\mathbb{C}} = \begin{pmatrix} A + iB & C + iD \\ -C + iD & A - iB \end{pmatrix}.$$

Compute $\alpha = \det H_{\mathbb{C}}$.
Output α.

(3.3.1) Theorem.

(1) *The output α is a non-negative real number and its expectation is the mixed discriminant $D(Q_1, \dots, Q_n)$.*

(2) *For any $C > 1$ the probability that*

$$\alpha \geq C \cdot D(Q_1, \dots, Q_n)$$

does not exceed C^{-1}.

(3) *For any $1 > \epsilon > 0$ the probability that*

$$\alpha \leq (\epsilon \mathfrak{c}_4)^n D(Q_1, \dots, Q_n)$$

does not exceed $8/(n \ln^2 \epsilon)$, *where* $\mathfrak{c}_4 \approx 0.76$ *is the constant defined in Section 1.3.*

(3.4) Relation to the permanent approximation algorithms. Let $A = (a_{ij})$ be an $n \times n$ non-negative matrix. Let us construct n matrices $Q_1, \dots, Q_n$ by placing the i-th row of A as the diagonal of Q_i: $Q_i = \operatorname{diag}\{a_{i1}, \dots, a_{in}\}$. Then we have $\operatorname{per} A = D(Q_1, \dots, Q_n)$ (cf. Section 1.2). Since A is non-negative, $Q_1, \dots, Q_n$ are positive semidefinite matrices. Now it is seen that if we choose $T_i = \operatorname{diag}\{\sqrt{a_{i1}}, \dots, \sqrt{a_{in}}\}$, then Algorithms 3.1–3.3 with the input $Q_1, \dots, Q_n$ transform into Algorithms 2.1–2.3 with the input A. Hence Theorems 3.1.1– 3.3.1 are straightforward generalizations of Theorems 2.1.1–2.3.1.

4. Gaussian Measures and Quadratic Forms

(4.1) Gaussian Measures. Given a space X with a probability measure μ, we denote the expectation of a function $f : X \longrightarrow \mathbb{R}$ by $\mathbf{E}(f)$. In this paper, X will be the space $\mathbb{R}^n$, $\mathbb{C}^n = \mathbb{R}^{2n}$, or $\mathbb{H}^n = \mathbb{R}^{4n}$ and μ will be a Gaussian measure (distribution).

Let $\|x\| = \sqrt{\xi_1^2 + \ldots + \xi_m^2}$ be the standard norm of a vector $x = (\xi_1, \ldots, \xi_m)$ in $\mathbb{R}^m$. As usual, we interpret $x \in \mathbb{R}^m$ as an m-column of numbers $\xi_1, \ldots, \xi_m$. The probability measure μ in $\mathbb{R}^m$ with the density $\psi(x) = (2\pi)^{-m/2} \exp\{-\|x\|^2/2\}$ is called the standard Gaussian (or Normal) distribution. A Gaussian distribution in $\mathbb{R}^m$ is the distribution of the vector Tx, where $x \in \mathbb{R}^m$ has the standard Gaussian distribution and T is a fixed $m \times m$ matrix (if T is degenerate, the distribution is concentrated on the image of T).

If μ is a probability distribution of $x = (\xi_1, \ldots, \xi_m) \in \mathbb{R}^m$, the matrix $Q = (q_{ij})$, where $q_{ij} = \mathbf{E}(\xi_i \xi_j)$, is called the *covariance matrix* of x. For example, if μ is the standard Gaussian distribution in $\mathbb{R}^m$, the covariance matrix is the $m \times m$ identity matrix I. We often use the following fact: if $x \in \mathbb{R}^m$ has the standard Gaussian distribution in $\mathbb{R}^m$ and T is an $m \times m$ matrix, then the covariance matrix of the vector Tx is $Q = TT^*$.

(4.2) Theorem. *Let us fix a Gaussian measure μ in $\mathbb{R}^n$. Let $q : \mathbb{R}^n \longrightarrow \mathbb{R}$ be a positive semidefinite quadratic form, such that*

$$\mathbf{E}(q) = 1.$$

Then

$$\mathfrak{C}_1 \leq \mathbf{E}(\ln q) \leq 0, \tag{1}$$

where $\mathfrak{C}_1 \approx -1.27$ is the constant defined in Section 1.3, and

$$0 \leq \mathbf{E}(\ln^2 q) \leq 8. \tag{2}$$

Proof. Without loss of generality, we can assume that μ is the standard Gaussian measure with the density

$$\psi(x) = (2\pi)^{-n/2} \exp\{-\|x\|^2/2\}$$

(otherwise, we apply a suitable linear transformation). Since ln is a concave function, by Jensen's inequality we have $\mathbf{E}(\ln q) \leq \ln \mathbf{E}(q) = 0$. Let us decompose q into a non-negative linear combination $q = \lambda_1 q_1 + \ldots + \lambda_n q_n$ of positive semidefinite forms q_i of rank 1. We can scale q_i so that $\mathbf{E}(q_i) = 1$ for $i = 1, \ldots, n$ and then we have $\lambda_1 + \ldots + \lambda_n = 1$. In fact, one can choose λ_i to be the eigenvalues of the matrix of q and $q_i = \langle x, u_i \rangle^2$, where u_i is the corresponding unit eigenvector and $\langle \cdot, \cdot \rangle$ is the standard scalar product in $\mathbb{R}^n$. Since ln is a concave function, we have $\ln(\lambda_1 q_1 + \ldots + \lambda_n q_n) \geq \lambda_1 \ln q_1 + \ldots + \lambda_n \ln q_n$. Furthermore, since q_i is a positive semidefinite form of rank 1, by an orthogonal transformation of the coordinates it can be brought into the form $q_i(x) = \alpha x_1^2$. Since $\mathbf{E}(x_1^2) = 1$, we conclude that $\alpha = 1$. Therefore, $\mathbf{E}(\ln q_i) = \mathbf{E}(\ln x_1^2) = \mathfrak{C}_1$ (cf. Sections 3.1 and 1.3) and

$$\mathbf{E}(\ln q) \geq \lambda_1 \mathbf{E}(\ln q_1) + \ldots + \lambda_n \mathbf{E}(\ln q_n) = (\lambda_1 + \ldots + \lambda_n)\mathfrak{C}_1 = \mathfrak{C}_1,$$

so Part 1 is proved (we note that this reasoning proves that $\mathbf{E}(\ln q)$ is well-defined).

Let $X = \{x \in \mathbb{R}^n : q(x) \leq 1\}$ and $Y = \mathbb{R}^n \setminus X$. Then

$$\mathbf{E}(\ln^2 q) = \int_X \psi(x) \ln^2 q(x)\ dx + \int_Y \psi(x) \ln^2 q(x)\ dx.$$

Let us estimate the first integral. Decomposing $q = \lambda_1 q_1 + \ldots + \lambda_n q_n$ as above, we get $\ln q \geq \lambda_1 \ln q_1 + \ldots + \lambda_n \ln q_n$. Since $\ln q(x) \leq 0$ for $x \in X$, we get that

$$\ln^2 q(x) \leq \sum_{i,j=1}^n \lambda_i \lambda_j \big(\ln q_i(x)\big)\big(\ln q_j(x)\big)$$

for $x \in X$. Therefore,

$$\int_X \psi(x) \ln^2 q(x)\ dx \leq \sum_{i,j=1}^n \lambda_i \lambda_j \int_X \psi(x) \big(\ln q_i(x)\big)\big(\ln q_j(x)\big)\ dx$$

$$\leq \sum_{i,j=1}^n \lambda_i \lambda_j \left(\int_X \psi(x) \ln^2 q_i(x)\ dx\right)^{1/2} \left(\int_X \psi(x) \ln^2 q_j(x)\ dx\right)^{1/2}$$

(we applied the Cauchy-Schwartz inequality)

$$\leq \sum_{i,j=1}^n \lambda_i \lambda_j \Big(\mathbf{E}(\ln^2 q_i)\Big)^{1/2} \Big(\mathbf{E}(\ln^2 q_j)\Big)^{1/2}.$$

Now, as in the proof of Part 1 we have

$$\mathbf{E}(\ln^2 q_i) = \mathbf{E}(\ln^2 x_1^2) = \frac{8}{\sqrt{2\pi}} \int_0^{+\infty} (\ln^2 t) e^{-t^2/2}\ dt \approx 6.548623960 \leq 7.$$

Summarizing, we get

$$\int_X \psi(x) \ln^2 q(x)\ dx \leq \sum_{i,j=1}^n \lambda_i \lambda_j \Big(\mathbf{E}(\ln^2 q_i)\Big)^{1/2} \Big(\mathbf{E}(\ln^2 q_j)\Big)^{1/2} \leq 7 \sum_{i,j=1}^n \lambda_i \lambda_j = 7.$$

Since for $0 \leq \ln t \leq \sqrt{t}$ for $t \geq 1$, we have

$$\int_Y \psi(x) \ln^2 q(x)\ dx \leq \int_Y q(x)\psi(x)\ dx \leq \mathbf{E}(q) = 1.$$

Therefore, $\mathbf{E}(\ln^2 q) \leq 7 + 1 = 8$ and Part 2 is proved. □

Remark. Role of the Gaussian distribution. Let us consider Algorithm 3.1. A natural question is to what extent it is important to sample vectors $u_1, \ldots, u_n$ from the Gaussian distribution, as compared to sampling from some other distribution μ in $\mathbb{R}^n$. Our method carries over as long as μ satisfies the following properties: first, the expectation of a random vector $x \in \mathbb{R}^n$ is 0 and the covariance matrix is the identity matrix I and second, if $q : \mathbb{R}^n \longrightarrow \mathbb{R}$ is a positive semidefinite quadratic form such that $\mathbf{E}(q) = 1$, then $\mathbf{E}(\ln q)$ is bounded below by a universal constant $\mathfrak{C} = \mathfrak{C}(\mu)$. Furthermore, the closer to 0 that $\mathfrak{C}$ can be chosen, the better approximation we get. It is seen that any discrete distribution (or, more generally, a distribution with atoms) fails the test for $n > 1$, since if a vector $x \in \mathbb{R}^n$ occurs with a positive probability and $q(x) = 0$, then $\mathbf{E}(\ln q) = -\infty$. One can use continuous distributions other than the Gaussian, but the author suspects that asymptotically, for large n, the Gaussian distribution provides the best constant $\mathfrak{C}$. For example, the uniform distribution on the sphere $\xi_1^2 + \ldots + \xi_n^2 = n$ in $\mathbb{R}^n$ might give better constants for small n, but asymptotically it gives the same constant $\mathfrak{C}_1$ (cf. Theorem 3.5 of [6]).

The next two results of this section state that for quadratic forms from some particular classes and some special Gaussian distributions we can get better estimates than in the general case.

Hermitian forms

We recall that a function $q : \mathbb{C}^n \longrightarrow \mathbb{R}$,

$$q(z) = \sum_{i,j=1}^{n} \beta_{ij} \zeta_i \overline{\zeta_j}, \quad \text{where} \quad z = (\zeta_1, \ldots, \zeta_n)$$

and $\beta_{ij} = \overline{\beta_{ji}}$ for $i, j = 1, \ldots, n$ is called a *Hermitian form* with the matrix (β_{ij}) (see, for example, Section 19 of [26]). The form is called a positive semidefinite, if $q(z) \geq 0$ for all $z \in \mathbb{C}^n$. We note that q can be considered as a real quadratic form $q : \mathbb{R}^{2n} \longrightarrow \mathbb{R}$, if we identify $\mathbb{C}^n = \mathbb{R}^{2n}$. We fix the standard complex Gaussian distribution in $\mathbb{C}^n$ with the density $\psi_{\mathbb{C}^n} = \dfrac{1}{\pi^n} e^{-\|z\|^2}$, cf. Section 3.2.

(4.3) Theorem. *Let us fix the standard complex Gaussian distribution in $\mathbb{C}^n$. Let $q : \mathbb{C}^n \longrightarrow \mathbb{R}$ be a positive semidefinite Hermitian form such that $\mathbf{E}(q) = 1$. Then*

$$\mathfrak{C}_2 \leq \mathbf{E}(\ln q) \leq 0,$$

where $\mathfrak{C}_2 \approx -0.58$ is the constant defined in Section 1.3.

Proof. The proof is similar to that of Part (1) of Theorem 4.2. Since q is a positive semidefinite Hermitian form, it can be represented as a convex combination $q = \lambda_1 q_1 + \ldots + \lambda_n q_n$ of positive semidefinite Hermitian forms q_i of rank 1 such that $\mathbf{E}(q_i) = 1$. Since ln is a concave function, we get that

$$\mathbf{E}(\ln q) \geq \lambda_1 \mathbf{E}(\ln q_1) + \ldots + \lambda_n \mathbf{E}(\ln q_n).$$

Since q_i is a positive semidefinite Hermitian form of rank 1, by a unitary transformation of $\mathbb{C}^n$, it can be brought into the form

$$q_i(z) = \alpha|\zeta_1|^2, \quad \text{where} \quad z = (\zeta_1, \ldots, \zeta_n)$$

and α is non-negative real. Since $\mathbf{E}(q_i) = 1$, we must have $\alpha = 1$, so $\mathbf{E}(\ln q_i) = \mathbf{E}(\ln|\zeta_1|^2) = \mathfrak{C}_2$ (cf. Sections 3.2 and 1.3). Summarizing, we get

$$\mathbf{E}(\ln q) \geq \lambda_1 \mathbf{E}(q_1) + \ldots + \lambda_n \mathbf{E}(q_n) = (\lambda_1 + \ldots + \lambda_n)\mathfrak{C}_2$$

and the proof follows. $\square$

As we see, the lower bound for the expectation of $\ln q$, where q is a Hermitian form, is better than for general quadratic forms. The reason for this improvement is, roughly, the following: the "worst possible" forms are those of rank 1. However, a Hermitian form of rank 1 is a *real* form of rank 2. Next, we will see that quaternionic quadratic forms provide a still better bound.

Quaternionic forms

Let $\mathbb{H}^n$ be the real vector space of all n-tuples $(\tau_1, \ldots, \tau_n)$ of quaternions $\tau_i \in \mathbb{H}$. As a vector space, $\mathbb{H}$ can be identified with $\mathbb{R}^4$ via the identification $a + \mathrm{i}\, b + \mathrm{j}\, c + \mathrm{k}\, d = (a, b, c, d)$, and so $\mathbb{H}^n$ can be identified with $\mathbb{R}^{4n}$. We fix the structure of a right $\mathbb{H}$-module of $\mathbb{H}^n$: for $u = (\tau_1, \ldots, \tau_n)$ and $\tau \in \mathbb{H}$, we let $u\tau = (\tau_1\tau, \ldots, \tau_n\tau)$. We note that the right multiplication by i, j, and k are orthogonal transformations of $\mathbb{R}^{4n}$ without fixed non-zero vectors. By a quadratic form on $\mathbb{H}^n$ we understand a function $q : \mathbb{H}^n \longrightarrow \mathbb{R}$ which is an ordinary quadratic form under the identification $\mathbb{H}^n = \mathbb{R}^{4n}$. We say that q is positive semidefinite if $q(u) \geq 0$ for any $u \in \mathbb{H}^n$. Let us fix the standard quaternionic Gaussian distribution in $\mathbb{H}^n$ with the density $\psi_{\mathbb{H}^n}(u) = \dfrac{4^n}{\pi^{2n}} e^{-2\|u\|^2}$, cf. Section 3.3.

(4.4) Theorem. *Let us fix the standard quaternionic Gaussian distribution in $\mathbb{H}^n$. Let $q : \mathbb{H}^n \longrightarrow \mathbb{R}$ be a positive semidefinite quadratic form such that $\mathbf{E}(q) = 1$. Suppose further, that $q(u) = q(u\,\mathrm{i}) = q(u\,\mathrm{j}) = q(u\,\mathrm{k})$ for any $u \in \mathbb{H}^n$. Then*

$$\mathfrak{C}_4 \leq \mathbf{E}(\ln q) \leq 0,$$

where $\mathfrak{C}_4 \approx -0.27$ is the constant defined in Section 1.3.

Proof. Let Q be the $4n \times 4n$ real symmetric matrix of $q : \mathbb{R}^{4n} \longrightarrow \mathbb{R}$ as a real quadratic form, so $q(x) = \langle Qx, x \rangle$, where $\langle \cdot, \cdot \rangle$ is the standard scalar product on $\mathbb{R}^{4n}$. Then for the differential of q at a point $x \in \mathbb{R}^{4n}$ we have $dq_x(\cdot) = 2\langle Qx, \cdot \rangle$. Let $S = \big\{x \in \mathbb{H}^n : \|x\| = 1\big\}$ be the unit sphere. As is known, $x \in S$ is an eigenvector of Q with an eigenvalue λ if and only if x is a critical point of the restriction $q : S \longrightarrow \mathbb{R}$ (that is, dq_x is 0 on the tangent space at x) with the corresponding critical value $\lambda = q(x)$.

Since q is invariant under the orthogonal transformations given by right multiplication by i, j, and k, we conclude that if x is an eigenvector of Q with an eigenvalue λ, then so are $x\,\mathrm{i}$, $x\,\mathrm{j}$, and $x\,\mathrm{k}$. It follows that each eigenspace of Q is a right $\mathbb{H}$-submodule of $\mathbb{H}^n$. In particular, the multiplicity of each eigenvalue of Q is a multiple of 4. Therefore, q can be expressed as a non-negative linear combination $q = \lambda_1 q_1 + \ldots + \lambda_n q_n$ of quadratic forms, such that for each i, $\mathbf{E}(q_i) = 1$ and by an orthogonal transformation of $\mathbb{R}^{4n}$, q_i can be written as a normalized sum of four squared coordinates: $q_i(x) = \alpha(\xi_1^2 + \xi_2^2 + \xi_3^2 + \xi_4^2)$. Since $\mathbf{E}(q_i) = 1$ we have $\alpha = 1$ and $\mathbf{E}(\ln q_i) = \mathfrak{C}_4$ (cf. Sections 3.3 and 1.3). Since ln is a concave function, we conclude that

$$\mathbf{E}(\ln q) \geq \lambda_1 \mathbf{E}(\ln q_1) + \ldots + \lambda_n \mathbf{E}(\ln q_n) \geq (\lambda_1 + \ldots + \lambda_n)\mathfrak{C}_4$$

and the proof follows. □

5. A Martingale Inequality

In this section, we prove a general result from the probability theory. Although the technique is quite standard, we present a complete proof here, since we need our estimate in a particular form suitable for proving the main results of the paper.

(5.1) Conditional expectations. In this subsection, we summarize some general results on measures and integration, which we exploit in Lemma 5.2 below. As a general source, we use [4].

Let us fix a probability measure μ on the Euclidean space $\mathbb{R}^m$. Suppose that μ is absolutely continuous with respect to the Lebesgue measure and let $\psi(x)$ be the density of μ. Suppose that we have k copies of the Euclidean space $\mathbb{R}^m$, each endowed with the measure μ. We consider functions $f : \mathbb{R}^m \times \ldots \times \mathbb{R}^m \longrightarrow \mathbb{R}$ that are defined almost everywhere and integrable with respect to the measure $\nu_k = \mu \times \ldots \times \mu$. Let $f(u_1, \ldots, u_k)$ be such a function. Then for almost all $(k-1)$-tuples $(u_1, \ldots, u_{k-1})$ with $u_i \in \mathbb{R}^m$, the function $f(u_1, \ldots, u_{k-1}, \cdot)$ is integrable (Fubini's Theorem) and we can define the *conditional expectation* $\mathbf{E}_k(f)$, which is a function of the first $k-1$ variables $u_1, \ldots, u_{k-1}$:

$$\mathbf{E}_k(f)(u_1, \ldots, u_{k-1}) = \int_{\mathbb{R}^m} f(u_1, \ldots, u_{k-1}, u_k)\psi(u_k)\ du_k.$$

Fubini's theorem implies that

$$\mathbf{E}(f) = \mathbf{E}_1 \ldots \mathbf{E}_k(f),$$

where $\mathbf{E}$ is the expectation with respect to the product measure ν_k. Tonelli's Theorem states that if f is ν_k-measurable and non-negative almost surely with respect to ν_k and if $\mathbf{E}_1 \ldots \mathbf{E}_k(f) < +\infty$, then f is ν_k-integrable.

If $f(u_1, \ldots, u_i)$ is a function of $i < k$ arguments, we may formally extend it to $\mathbb{R}^m \times \ldots \times \mathbb{R}^m$ (k times) by letting $f(u_1, \ldots, u_k) = f(u_1, \ldots, u_i)$. If $f(u_1, \ldots, u_i)$ is ν_i-integrable, then $f(u_1, \ldots, u_k)$ is ν_k-integrable.

We note the following useful facts:

(5.1.1) The operator $\mathbf{E}_k$ is linear and *monotone*, that is,
if $f(u_1, \ldots, u_k) \leq g(u_1, \ldots, u_k)$ almost surely with respect to ν_k,
then $\mathbf{E}_k(f) \leq \mathbf{E}_k(g)$ almost surely with respect to ν_{k-1}.

(5.1.2) If $f(u_1, \ldots, u_k)$ is integrable and $g(u_1, \ldots, u_i)$, $i < k$ is a ν_i-measurable function, then $\mathbf{E}_k(gf) = g\mathbf{E}_k(f)$.

(5.1.3) If $f = a$ is a constant almost surely with respect to ν_k,
then $\mathbf{E}_k(f) = a$ almost surely with respect to ν_{k-1}.

In this section, we prove the following technical lemma (a martingale inequality).

(5.2) Lemma. *Suppose that $f_k(u_1, \ldots, u_k)$, $k = 1, \ldots, n$ is an integrable function on the product $\mathbb{R}^m \times \ldots \times \mathbb{R}^m$ of k copies of $\mathbb{R}^m$, and let $\nu_n = \mu \times \ldots \times \mu$ be the n-th product measure. Suppose that for some $a, b \in \mathbb{R}$ and all $k = 1, \ldots, n$*

$$a \leq \mathbf{E}_k(f_k) \quad \textit{and} \quad \mathbf{E}_k(f_k^2) \leq b \qquad \textit{almost surely with respect to} \quad \nu_{k-1}.$$

Then for any $\delta > 0$,

$$\nu_n\Big\{(u_1, \ldots, u_n) : \ \frac{1}{n}\sum_{k=1}^n f_k(u_1, \ldots, u_k) \leq a - \delta\Big\} \leq \frac{b}{n\delta^2}.$$

Proof. Let $g_k = \mathbf{E}_k(f_k)$ and let $h_k = f_k - g_k$. Since g_k does not depend on u_k, using (5.1.2), we get

$$\mathbf{E}_k(h_k^2) = \mathbf{E}_k(f_k^2) - 2\mathbf{E}_k(g_k f_k) + \mathbf{E}_k(g_k^2) = \mathbf{E}_k(f_k^2) - g_k^2.$$

Hence we may write

$$f_k = g_k + h_k, \quad \text{where} \quad \mathbf{E}_k(h_k) = 0, \quad g_k \geq a, \qquad \text{and} \qquad \mathbf{E}_k(h_k^2) \leq b$$

almost surely with respect to ν_{k-1}. Let

$$H(u_1, \ldots, u_n) = \frac{1}{n}\sum_{k=1}^n h_k(u_1, \ldots, u_k).$$

Now for $U = (u_1, \ldots, u_n)$ we have

$$\nu_n\Big\{U : \frac{1}{n}\sum_{k=1}^n f_k(u_1, \ldots, u_k) \leq a - \delta\Big\}$$

$$= \nu_n\Big\{U : \ H(U) + \frac{1}{n}\sum_{k=1}^n g_k(u_1, \ldots, u_{k-1}) \leq a - \delta\Big\}$$

$$\le \nu_n\Big\{U:\ H(U) \le -\delta\Big\} \le \frac{\mathbf{E}(H^2)}{\delta^2} \quad \text{(we used Chebyshev's inequality)}$$

$$= \frac{1}{\delta^2 n^2}\sum_{k=1}^n \mathbf{E}(h_k^2) + \frac{2}{\delta^2 n^2}\sum_{1\le i<j\le n} \mathbf{E}(h_i h_j).$$

We note that it is legitimate to pass to global expectations $\mathbf{E}$ here. Indeed, since h_k^2 is non-negative and $\mathbf{E}_k h_k^2 \le \mathbf{E}_k f_k^2 \le b$ it follows by formulas (5.1.1), (5.1.3) and Tonelli's Theorem that h_k^2 is ν_n-integrable. Since $|h_i h_j| \le (h_i^2 + h_j^2)/2$, the products $h_i h_j$ are also ν_n-integrable. Therefore, H^2 is ν_n-integrable.

Since h_k does not depend on $u_{k+1}, \dots, u_n$, using (5.1.2), we have $\mathbf{E}(h_k^2) = \mathbf{E}_1 \dots \mathbf{E}_n h_k^2 = \mathbf{E}_1 \dots \mathbf{E}_k h_k^2$ and since $\mathbf{E}_k h_k^2 \le b$ almost surely on ν_{k-1}, by (5.1.1) and (5.1.3) we get that $\mathbf{E}(h_k^2) \le b$ for each $k = 1, \dots, n$. Furthermore, since h_i does not depend on $u_{i+1}, \dots, u_n$, using formulas (5.1.2) and (5.1.3) we get that for $j > i$

$$\mathbf{E}(h_i h_j) = \mathbf{E}_1 \dots \mathbf{E}_n(h_i h_j) = \mathbf{E}_1 \dots \mathbf{E}_j(h_i h_j) = \mathbf{E}_i \dots \mathbf{E}_{j-1} h_i \mathbf{E}_j(h_j) = 0.$$

The proof now follows. □

6. Proofs

As we noted in Section 3.4, Theorems 3.1.1–3.3.1 imply Theorems 2.1.1–2.3.1. Proofs of Theorems 3.1.1, 3.2.1, and 3.3.1 are very similar. Theorem 3.3.1 provides the best approximation known so far and it is the hardest to prove, so we present its detailed proof here. Theorem 2.1.1 is the easiest to prove and we present its detailed proof as well, because there the main ideas of the general method can be easily traced. Then we describe the modifications one needs to make to prove Theorems 2.2.1, 3.1.1, and 3.2.1.

Proof of Theorem 2.1.1. The proof of Part 1 follows the proof that the Godsil-Gutman estimator is unbiased (see Chapter 8 of [23]). Let us write α as a polynomial in u_{ij}:

$$\alpha = (\det B)^2 = \left(\sum_{\sigma\in S_n} \operatorname{sgn}\,\sigma \prod_{i=1}^n u_{i\sigma(i)}\sqrt{a_{i\sigma(i)}}\right)^2$$

$$= \sum_{\sigma_1,\sigma_2\in S_n} (\operatorname{sgn}\,\sigma_1)(\operatorname{sgn}\,\sigma_2)\prod_{i=1}^n u_{i\sigma_1(i)}u_{i\sigma_2(i)}\sqrt{a_{i\sigma_1(i)}a_{i\sigma_2(i)}}.$$

For every pair of permutations $\sigma_1, \sigma_2 \in S_n$, the corresponding summand is a monomial in variables u_{ij}. Each variable u_{ij} from the monomial occurs either with degree 2 if $\sigma_1(i) = \sigma_2(i) = j$, or with degree 1 otherwise. Next, we observe that unless $\sigma_1 = \sigma_2$, the corresponding summand contains some u_{ij} with degree 1. Since the expectation of u_{ij} is 0, the expectation of the whole monomial is 0. Hence,

$$\mathbf{E}(\alpha) = \sum_{\sigma\in S_n} (\operatorname{sgn}\ \sigma)^2 \mathbf{E}\left(\prod_{i=1}^n u_{i\sigma(i)}^2 a_{i\sigma(i)}\right).$$

Since u_{ij} are independent and $\mathbf{E}(u_{ij})^2 = 1$, we conclude that $\mathbf{E}(\alpha) = \text{per } A$.

Part 2 follows from Part 1, the non-negativity of α, and the Chebyshev inequality.

To prove Part 3, let us introduce vectors $u_1, \ldots, u_n \in \mathbb{R}^n$, where

$$u_i = (u_{i1}, \ldots, u_{in})$$

is the i-th row of the matrix u_{ij}. Thus $u_1, \ldots, u_n$ are vectors sampled independently from the standard Gaussian distribution in $\mathbb{R}^n$ and the output $\alpha = \alpha(u_1, \ldots, u_n)$ is a function of $u_1, \ldots, u_n$,

$$\alpha : \mathbb{R}^n \times \ldots \times \mathbb{R}^n \longrightarrow \mathbb{R}.$$

Since $\alpha = (\det B)^2$ and the determinant is a linear function in every row, we conclude that for each $i = 1, \ldots, n$, the output α is a quadratic form in $u_i \in \mathbb{R}^n$, provided $u_1, \ldots, u_{i-1}, u_{i+1}, \ldots, u_n$ are fixed. Furthermore, since $\alpha \geq 0$, we deduce that α is a positive semidefinite quadratic form in u_i.

As in Section 5.1, let us introduce the conditional expectation $\mathbf{E}_k$ with respect to u_k, $k = 1, \ldots, n$. Hence we can write

$$\text{per } A = \mathbf{E}(\alpha) = \mathbf{E}_1 \ldots \mathbf{E}_n \alpha(u_1, \ldots, u_n).$$

Let $\alpha_k(u_1, \ldots, u_k) = \mathbf{E}_{k+1} \ldots \mathbf{E}_n(\alpha)$. Thus α_k is a polynomial function in the first k vectors $u_1, \ldots, u_k$, and α_k is a positive semidefinite quadratic form in u_k, provided $u_1, \ldots, u_{k-1}$ are fixed. Naturally, $\alpha_0 = \text{per } A$ and $\alpha_n = \alpha$. Without loss of generality, we may assume that per $A > 0$, for if per $A = 0$, then by Part 2, $\alpha = 0$ almost surely and Part 3 is obvious. Hence $\alpha_k(u_1, \ldots, u_k) > 0$ for almost all k-tuples $(u_1, \ldots, u_k)$.

We may write

$$\frac{\alpha}{\text{per } A} = \prod_{k=1}^{n} \frac{\alpha_k(u_1, \ldots, u_k)}{\alpha_{k-1}(u_1, \ldots, u_{k-1})},$$

and, therefore,

$$\frac{1}{n} \ln \frac{\alpha}{\text{per } A} = \frac{1}{n} \sum_{i=1}^{n} \ln \frac{\alpha_k(u_1, \ldots, u_k)}{\alpha_{k-1}(u_1, \ldots, u_{k-1})}.$$

We observe that for each fixed $(k-1)$-tuple $(u_1, \ldots, u_{k-1})$ such that $\alpha_{k-1}(u_1, \ldots, u_{k-1}) \neq 0$, the ratio

$$\frac{\alpha_k(u_1, \ldots, u_k)}{\alpha_{k-1}(u_1, \ldots, u_{k-1})}$$

is a positive semidefinite quadratic form in u_k whose expectation is 1. Therefore, by Theorem 4.2,

$$\mathfrak{C}_1 \leq \mathbf{E}_k \left(\ln \frac{\alpha_k(u_1, \ldots, u_k)}{\alpha_{k-1}(u_1, \ldots, u_{k-1})} \right)$$

and

$$\mathbf{E}_k \left(\ln^2 \frac{\alpha_k(u_1, \ldots, u_k)}{\alpha_{k-1}(u_1, \ldots, u_{k-1})} \right) \leq 8.$$

Now we apply Lemma 5.2 with

$$f_k(u_1, \ldots, u_k) = \ln \frac{\alpha_k(u_1, \ldots, u_k)}{\alpha_{k-1}(u_1, \ldots, u_{k-1})}, \quad a = \mathfrak{C}_1, \quad b = 8, \quad \text{and} \quad \delta = -\ln \epsilon$$

to conclude that for any $1 > \epsilon > 0$

$$\text{Probability}\left\{ \frac{1}{n} \ln \frac{\alpha}{\text{per } A} \leq \mathfrak{C}_1 + \ln \epsilon \right\} \leq \frac{8}{n \ln^2 \epsilon},$$

and, since $\mathfrak{c}_1 = \exp\{\mathfrak{C}_1\}$ (see Section 1.3),

$$\text{Probability}\left\{ \alpha \leq \left(\mathfrak{c}_1 \epsilon\right)^n \text{per } A \right\} \leq \frac{8}{n \ln^2 \epsilon}.$$

The proof of Part 3 now follows. □

As we see, the proof is based on the following three observations:

First, the output α is a non-negative number.

Second, the expectation of α is the value we are seeking to approximate.

Third, α can be represented as a function $\alpha(u_1, \ldots, u_n)$ of vectors u_i, drawn independently from a Gaussian distribution in $\mathbb{R}^n$, so that for any fixed $u_1, \ldots, u_{i-1}$, $u_{i+1}, \ldots, u_n$, the function α is a quadratic form in u_i.

To obtain the proof of Theorem 3.1.1, we need to do some minor modifications. First, we note that it is clear that α is non-negative and that $\alpha(u_1, \ldots, u_n)$ is a quadratic form in u_i, provided $u_1, \ldots, u_{i-1}, u_{i+1}, \ldots, u_n$ are fixed. The proof that α provides an unbiased estimator is very similar to that of Theorem 2.1.1. Let $w_k = T_k u_k$, $w_k = (\omega_{k1}, \ldots, \omega_{kn})$. Then the covariance matrix of w_k is $T_k T_k^* = Q_k$ (cf. Section 4.1), so $\mathbf{E}(\omega_{ki}\omega_{kj}) = q_{ij,k}$, where $Q_k = (q_{ij,k})$. Furthermore, vectors w_i and w_j are independent for $i \neq j$. Now

$$\mathbf{E}(\alpha) = \mathbf{E}\Big(\det[w_1, \ldots, w_n]\Big)^2 = \mathbf{E}\left(\sum_{\sigma \in S_n} \text{sgn } \sigma \prod_{k=1}^n \omega_{k\sigma(k)} \right)^2$$

$$= \mathbf{E}\left(\sum_{\sigma_1, \sigma_2 \in S_n} (\text{sgn } \sigma_1)(\text{sgn } \sigma_2) \prod_{k=1}^n \omega_{k\sigma_1(k)} \omega_{k\sigma_2(k)} \right)$$

$$= \sum_{\sigma_1, \sigma_2 \in S_n} (\text{sgn } \sigma_1)(\text{sgn } \sigma_2) \prod_{k=1}^n \mathbf{E}\Big(\omega_{k\sigma_1(k)} \omega_{k\sigma_2(k)}\Big)$$

$$= \sum_{\sigma_1, \sigma_2 \in S_n} (\text{sgn } \sigma_1)(\text{sgn } \sigma_2) \prod_{k=1}^{n} q_{\sigma_1(k)\sigma_2(k),k} = D(Q_1, \ldots, Q_n)$$

by (1.2.1).

To prove Theorems 2.2.1 and 3.2.1, we note that, if $u_1, \ldots, u_{i-1}, u_{i+1}, \ldots, u_n \in \mathbb{C}^n$ are fixed, α is a Hermitian form in $u_i \in \mathbb{C}^n$, so instead of Part 1 of Theorem 4.2 one should refer to Theorem 4.3.

The proof of Theorem 3.3.1 is much simplified if we use the exterior algebra formalism (see, for example, Section 28 of [26]).

(6.1) Exterior Algebra. Let V be a complex vector space and suppose that $\dim\ V = m$. Recall that the exterior algebra $\bigwedge V$, as a vector space, is the direct sum $\bigwedge V = \bigoplus_{k=1}^{m} \bigwedge^k V$, where $\bigwedge^k V$ is spanned by *wedge products*

$$v_1 \wedge \ldots \wedge v_k, \quad v_i \in V,\ i = 1, \ldots, k,$$

which are linear in each argument

$$v_1 \wedge \ldots \wedge v_{i-1} \wedge (\alpha v_i' + \beta v_i'') \wedge v_{i+1} \wedge \ldots \wedge v_k$$

$$= \alpha(v_1 \wedge \ldots \wedge v_{i-1} \wedge v_i' \wedge v_{i+1} \wedge \ldots \wedge v_k) + \beta(v_1 \wedge \ldots \wedge v_{i-1} \wedge v_i'' \wedge v_{i+1} \wedge \ldots \wedge v_k)$$

and skew-symmetric:

$$v_{\sigma(1)} \wedge \ldots \wedge v_{\sigma(k)} = (\text{sgn } \sigma)(v_1 \wedge \ldots \wedge v_k)$$

for any permutation $\sigma \in S_k$. Vectors from $\bigwedge^k V$ are called k-vectors.

Let us fix a basis $e_1, \ldots, e_m$ of V, thus identifying $V = \mathbb{C}^m$. For an $n \times n$ matrix U with the columns $u_1, \ldots, u_m \in \mathbb{C}^m$ we have

$$u_1 \wedge \ldots \wedge u_m = (\det U)(e_1 \wedge \ldots \wedge e_m).$$

We identify $\bigwedge^m(\mathbb{C}^m) = \mathbb{C}$, so we write simply

$$u_1 \wedge \ldots \wedge u_m = \det U.$$

Let $x = (\xi_1, \ldots, \xi_m), y = (\eta_1, \ldots, \eta_m) \in \mathbb{C}^m$. Then

$$x \wedge y = \sum_{1 \le i < j \le m} (\xi_i \eta_j - \xi_j \eta_i)(e_i \wedge e_j). \tag{6.1.1}$$

(6.2) Proposition. *For a vector $u \in \mathbb{H}^n$, let us define a 2-vector $\omega(u) \in \bigwedge^2 \mathbb{C}^{2n}$ as follows: if $u = a + \mathrm{i}\, b + \mathrm{j}\, c + \mathrm{k}\, d$ with $a, b, c, d \in \mathbb{R}^n$, we let*

$$\omega(u) = (a + ib, -c + id) \wedge (c + id, a - ib).$$

Then

(1) *For any $u \in \mathbb{H}^n$, $\omega(u) = \omega(u\,\mathrm{i}) = \omega(u\,\mathrm{j}) = \omega(u\,\mathrm{k})$.*

(2) *Suppose that H is a quaternionic $n \times n$ matrix with the columns $u_1, \ldots, u_n \in \mathbb{H}^n$. Let us write $H = A + \mathrm{i}\, B + \mathrm{j}\, C + \mathrm{k}\, D$ for $n \times n$ real matrices A, B, C, and D and let $H_{\mathbb{C}}$ be the $2n \times 2n$ complex matrix*

$$H_{\mathbb{C}} = \begin{pmatrix} A + iB & C + iD \\ -C + iD & A - iB \end{pmatrix}.$$

Then

$$\det H_{\mathbb{C}} = (-1)^{\frac{n(n-1)}{2}} \omega(u_1) \wedge \ldots \wedge \omega(u_n).$$

(3) *Let T be an $n \times n$ real matrix and let $Q = TT^*$, $Q = (q_{ij})$. Suppose that u is sampled from the standard quaternionic Gaussian distribution in $\mathbb{H}^n$. Then the expectation of $\omega(Tu)$ is the 2-vector*

$$\sum_{i,j=1}^{n} q_{ij} (e_i \wedge e_{j+n}),$$

where $e_1, \ldots, e_{2n}$ is the standard basis of $\mathbb{C}^{2n}$.

Proof. Part 1 is proved by direct computation. Let us denote $v_1 = (a+ib, -c+id)$, $v_2 = (c + id, a - ib)$, so $\omega(u) = v_1 \wedge v_2$. Then $\omega(u\,\mathrm{i}) = (iv_1) \wedge (-iv_2) = v_1 \wedge v_2$, $\omega(u\,\mathrm{j}) = (-v_2) \wedge v_1 = v_1 \wedge v_2$, and $\omega(u\,\mathrm{k}) = (iv_2) \wedge (iv_1) = -(v_2 \wedge v_1) = v_1 \wedge v_2$.

It is convenient to think of vectors as columns of numbers, so we write

$$\omega(u) = \begin{pmatrix} a + ib \\ -c + id \end{pmatrix} \wedge \begin{pmatrix} c + id \\ a - ib \end{pmatrix}$$

for $u = a + \mathrm{i}\, b + \mathrm{j}\, c + \mathrm{k}\, d$.

To prove Part 2, let us denote the k-th column of A, B, C, and D by a_k, b_k, c_k, and d_k respectively. Then

$$\det \begin{pmatrix} A + iB & C + iD \\ -C + iD & A - iB \end{pmatrix}$$

$$= \begin{pmatrix} a_1 + ib_1 \\ -c_1 + id_1 \end{pmatrix} \wedge \ldots \wedge \begin{pmatrix} a_n + ib_n \\ -c_n + id_n \end{pmatrix} \wedge \begin{pmatrix} c_1 + id_1 \\ a_1 - ib_1 \end{pmatrix} \wedge \ldots \wedge \begin{pmatrix} c_n + id_n \\ a_n - ib_n \end{pmatrix}.$$

Rearranging the vectors in the wedge product, we get

$$\det \begin{pmatrix} A + iB & C + iD \\ -C + iD & A - iB \end{pmatrix}$$

$$= (-1)^{\frac{n(n-1)}{2}} \begin{pmatrix} a_1 + ib_1 \\ -c_1 + id_1 \end{pmatrix} \wedge \begin{pmatrix} c_1 + id_1 \\ a_1 - ib_1 \end{pmatrix} \wedge \ldots \wedge \begin{pmatrix} a_n + ib_n \\ -c_n + id_n \end{pmatrix} \wedge \begin{pmatrix} c_n + id_n \\ a_n - ib_n \end{pmatrix}$$

$$= (-1)^{\frac{n(n-1)}{2}} \omega(u_1) \wedge \ldots \wedge \omega(u_n).$$

To prove Part 3, let $u = a + \mathrm{i}\, b + \mathrm{j}\, c + \mathrm{k}\, d$ for $a, b, c, d \in \mathbb{R}^n$. Then $Tu = Ta + \mathrm{i}\, Tb + \mathrm{j}\, Tc + \mathrm{k}\, Td$ and

$$\omega(Tu) = \begin{pmatrix} Ta + iTb \\ -Tc + iTd \end{pmatrix} \wedge \begin{pmatrix} Tc + iTd \\ Ta - iTb \end{pmatrix}.$$

Let $Ta = (\alpha_1, \ldots, \alpha_n)$, $Tb = (\beta_1, \ldots, \beta_n)$, $Tc = (\gamma_1, \ldots, \gamma_n)$ and $Td = (\delta_1, \ldots, \delta_n)$ be the coordinates of Ta, Tb, Tc, and Td respectively. Since a, b, c, and d are sampled independently from the Gaussian distribution in $\mathbb{R}^n$ with the covariance matrix $(1/4)I$, it follows that Ta, Tb, Tc, and Td are sampled from the distribution with the covariance matrix $(1/4)Q$ (see Section 4.1), so

$$\mathbf{E}(\alpha_i \alpha_j) = \mathbf{E}(\beta_i \beta_j) = \mathbf{E}(\gamma_i \gamma_j) = \mathbf{E}(\delta_i \delta_j) = \frac{q_{ij}}{4}$$

and all other pairs from the set $\alpha_1, \beta_1, \gamma_1, \delta_1, \ldots, \alpha_n, \beta_n, \gamma_n, \delta_n$ are uncorrelated. Using (6.1.1), we can write

$$\omega(Tu) = \mathrm{I} + \mathrm{II} + \mathrm{III}, \quad \text{where}$$

$$\mathrm{I} = \sum_{1 \le k < s \le n} \big((\alpha_k + i\beta_k)(\gamma_s + i\delta_s) - (\alpha_s + i\beta_s)(\gamma_k + i\delta_k) \big) \big(e_k \wedge e_s \big),$$

$$\mathrm{II} = \sum_{1 \le k, s \le n} \big((\alpha_k + i\beta_k)(\alpha_s - i\beta_s) - (-\gamma_s + i\delta_s)(\gamma_k + i\delta_k) \big) \big(e_k \wedge e_{s+n} \big),$$

and

$$\mathrm{III} = \sum_{1 \le k < s \le n} \big((-\gamma_k + i\delta_k)(\alpha_s - i\beta_s) - (-\gamma_s + i\delta_s)(\alpha_k - i\beta_k) \big) \big(e_{k+n} \wedge e_{s+n} \big).$$

It seen that the expectations of the first and the last sum are 0, whereas the expectation of the second sum is

$$\sum_{1 \le k, s \le n} q_{ks} \big(e_k \wedge e_{s+n} \big).$$

□

Proof of Theorem 3.3.1. The output α is a non-negative real number, since it is the reduced norm (squared Dieudonné determinant) of a quaternionic matrix (see Chapter IV, Section 1 of [3]). A direct proof of this fact is as follows (cf. [3]). One can observe that $\det H_{\mathbb{C}} = \det \overline{H}_{\mathbb{C}}$, so $\alpha = \det H_{\mathbb{C}}$ is a real number. The correspondence $H \longmapsto H_{\mathbb{C}}$ is an embedding of the group $GL_n(\mathbb{H})$ of the non-degenerate

$n \times n$ quaternionic matrices in the group $GL_{2n}(\mathbb{C})$ of $2n \times 2n$ non-degenerate complex matrices. The group $GL_n(\mathbb{H})$ is known to be connected, therefore $\det H_{\mathbb{C}}$ can not change sign as H changes within the group. Substituting the identity matrix $H = I$, we conclude that $\det H_{\mathbb{C}}$ is positive for any $H \in GL_n(\mathbb{H})$. Since the group $GL_n(\mathbb{H})$ is dense in the space of all $n \times n$ quaternionic matrices H, we conclude that the output α is non-negative.

Let us prove that the expectation of α is the mixed discriminant $D(Q_1, \ldots, Q_n)$. Applying Part 2 of Proposition 6.2, we can write

$$\alpha = \alpha(u_1, \ldots, u_n) = (-1)^{\frac{n(n-1)}{2}} \omega(T_1 u_1) \wedge \ldots \wedge \omega(T_n u_n).$$

Let $\mathbf{E}_k$ be the conditional expectation with respect to $u_k \in \mathbb{H}^n$. Since the wedge product is linear in every term, we may write

$$\mathbf{E}(\alpha) = \mathbf{E}_1 \ldots \mathbf{E}_n(\alpha) = (-1)^{\frac{n(n-1)}{2}} \mathbf{E}_1\big(\omega(T_1 u_1)\big) \wedge \ldots \wedge \mathbf{E}_n\big(\omega(T_n u_n)\big).$$

Let $Q_k = (q_{ij,k})$ for $k = 1, \ldots, n$ and $1 \leq i, j \leq n$. Applying Part 3 of Proposition 6.2, we get

$$\mathbf{E}(\alpha) = (-1)^{\frac{n(n-1)}{2}} \left(\sum_{i,j=1}^{n} q_{ij,1}(e_i \wedge e_{j+n}) \right) \wedge \ldots \wedge \left(\sum_{i,j=1}^{n} q_{ij,n}(e_i \wedge e_{j+n}) \right).$$

Rearranging the terms in the wedge product and canceling wedge products containing repeated vectors, we get

$$\begin{aligned}
&\mathbf{E}(\alpha) \\
&= (-1)^{\frac{n(n-1)}{2}} \sum_{1 \leq i_1, \ldots, i_n; j_1, \ldots, j_n \leq n} \left(\prod_{k=1}^{n} q_{i_k j_k, k} \right) e_{i_1} \wedge e_{j_1+1} \wedge \ldots \wedge e_{i_n} \wedge e_{j_n+n} \\
&= (-1)^{\frac{n(n-1)}{2}} \sum_{\sigma_1, \sigma_2 \in S_n} \left(\prod_{k=1}^{n} q_{\sigma_1(k)\sigma_2(k),k} \right) e_{\sigma_1(1)} \wedge e_{\sigma_2(1)+1} \wedge \ldots \wedge e_{\sigma_1(n)} \wedge e_{\sigma_2(n)+n} \\
&= (-1)^{\frac{n(n-1)}{2}} \sum_{\sigma_1, \sigma_2 \in S_n} (\operatorname{sgn} \ \sigma_1)(\operatorname{sgn} \ \sigma_2) \left(\prod_{k=1}^{n} q_{\sigma_1(k)\sigma_2(k),k} \right) e_1 \wedge e_{n+1} \wedge \ldots \wedge e_n \wedge e_{2n} \\
&= \sum_{\sigma_1, \sigma_2 \in S_n} (\operatorname{sgn} \ \sigma_1)(\operatorname{sgn} \ \sigma_2) \left(\prod_{k=1}^{n} q_{\sigma_1(k)\sigma_2(k),k} \right) e_1 \wedge \ldots \wedge e_n \wedge e_{n+1} \wedge \ldots \wedge e_{2n} \\
&= D(Q_1, \ldots, Q_n) \quad \text{by (1.2.1)}
\end{aligned}$$

and Part 1 is proven.

Part 2 follows by Part 1 and the Chebyshev inequality.

Let us prove Part 3. As in the proof of Theorem 2.1.1, we introduce conditional expectations

$$\alpha_k(u_1, \ldots, u_k) = \mathbf{E}_{k+1} \ldots \mathbf{E}_n(\alpha)$$

$$= (-1)^{\frac{n(n-1)}{2}} \omega(T_1 u_1) \wedge \ldots \wedge \omega(T_k u_k) \wedge \mathbf{E}_{k+1}\big(\omega(T_{k+1} u_{k+1})\big) \wedge \ldots \wedge \mathbf{E}_n\big(\omega(T_n u_n)\big).$$

We have $\alpha_0 = \alpha$ and $\alpha_n = D(Q_1, \ldots, Q_n)$.

Since the wedge product is linear in every term, we conclude that for any fixed $u_1, \ldots, u_{k-1}$, the function $\alpha_k(u_1, \ldots, u_{k-1}, u_k)$ is a (necessarily positive semidefinite) quadratic form in $u_k \in \mathbb{H}^n$. Furthermore, since the multiplications by a real matrix T and the quaternion units i, j, and k commute, by Part 1 of Proposition 6.2 we conclude that $\alpha_k(u_1, \ldots, u_{k-1}, u_k)$ is invariant under the right multiplication of u_k by i, j, and k.

Without loss of generality, we may suppose that $D(Q_1, \ldots, Q_n) > 0$, since if $D(Q_1, \ldots, Q_n) = 0$, by Part 1 we have $\alpha = 0$ almost surely and the proof would follow immediately. Since $\alpha_k(u_1, \ldots, u_k)$ is a polynomial in $u_1, \ldots, u_k$, we conclude that $\alpha_k(u_1, \ldots, u_k) > 0$ almost surely.

We may write

$$\frac{\alpha}{D(Q_1, \ldots, Q_n)} = \prod_{k=1}^{n} \frac{\alpha_k(u_1, \ldots, u_k)}{\alpha_{k-1}(u_1, \ldots, u_{k-1})}.$$

and, therefore,

$$\frac{1}{n} \ln \frac{\alpha}{D(Q_1, \ldots, Q_n)} = \frac{1}{n} \sum_{i=1}^{n} \ln \frac{\alpha_k(u_1, \ldots, u_k)}{\alpha_{k-1}(u_1, \ldots, u_{k-1})}.$$

We observe that for each fixed $(k-1)$-tuple $(u_1, \ldots, u_{k-1})$, such that $\alpha_{k-1}(u_1, \ldots, u_{k-1}) \neq 0$, the ratio

$$\frac{\alpha_k(u_1, \ldots, u_k)}{\alpha_{k-1}(u_1, \ldots, u_{k-1})}$$

is a positive semidefinite quadratic form in u_k, which is invariant under the right multiplication by i, j, and k and which has expectation 1. Therefore, by Theorem 4.4,

$$\mathfrak{C}_4 \leq \mathbf{E}_k \left(\ln \frac{\alpha_k(u_1, \ldots, u_k)}{\alpha_{k-1}(u_1, \ldots, u_{k-1})} \right),$$

and by Part 2 of Theorem 4.2,

$$\mathbf{E}_k \left(\ln^2 \frac{\alpha_k(u_1, \ldots, u_k)}{\alpha_{k-1}(u_1, \ldots, u_{k-1})} \right) \leq 8.$$

Now we apply Lemma 5.2 with

$$f_k(u_1, \ldots, u_k) = \ln \frac{\alpha_k(u_1, \ldots, u_k)}{\alpha_{k-1}(u_1, \ldots, u_{k-1})}, \quad a = \mathfrak{C}_4, \quad b = 8, \quad \text{and} \quad \delta = -\ln \epsilon$$

to conclude that for any $1 > \epsilon > 0$

$$\text{Probability}\left\{\frac{1}{n} \ln \frac{\alpha}{D(Q_1, \dots , Q_n)} \leq \mathfrak{C}_4 + \ln \epsilon\right\} \leq \frac{8}{n \ln^2 \epsilon},$$

and, since $\mathfrak{c}_4 = \exp\{\mathfrak{C}_4\}$ (cf. Section 1.3),

$$\text{Probability}\left\{\alpha \leq \left(\mathfrak{c}_4 \epsilon\right)^n D(Q_1, \dots , Q_n)\right\} \leq \frac{8}{n \ln^2 \epsilon}.$$

The proof of Part 3 now follows. □

7. Possible Ramifications

It appears that our method can be used in the following general situation. Suppose we want to approximate some quantity α_0 of interest. Suppose that we have a function

$$\alpha : \mathbb{R}^{m_1} \times \dots \times \mathbb{R}^{m_n} \longrightarrow \mathbb{R},$$

where each space $\mathbb{R}^{m_i}$ is endowed with a Gaussian probability measure μ_i. Suppose that the function $\alpha(u_1, \dots , u_n)$ has the following properties:

a) The value $\alpha(u_1, \dots , u_n)$ is a non-negative number, which can be efficiently computed for any given vectors $u_1 \in \mathbb{R}^{m_1}, \dots , u_n \in \mathbb{R}^{m_n}$.

b) For any fixed $u_1, \dots , u_{i-1}, u_{i+1}, \dots , u_n$, the value $\alpha(u_1, \dots , u_n)$ is a quadratic form in $u_i \in \mathbb{R}^{m_i}$.

c) The expectation of α with respect to the product measure $\mu_1 \times \dots \times \mu_n$ is the quantity α_0.

Then we get an efficient randomized algorithm to approximate α_0 within a simply exponential factor $O(c^n)$, where $1 > c > 0$ is an absolute constant. The algorithm consists of sampling $u_1, \dots , u_n$ independently and at random and computing $\alpha(u_1, \dots , u_n)$.

Example. Sums of subpermanents. Let $A = (a_{ij})$ be a rectangular $n \times m$ matrix, $m \geq n$. For a subset $I \subset \{1, \dots , m\}$ of the cardinality n, let A_I be the $n \times n$ submatrix of A consisting of the columns whose indices are in I. Let

$$\text{PER } A = \sum_{|I|=n} \text{per } \ A_I,$$

where the sum is taken over all subsets $I \subset \{1, \dots , m\}$ of the cardinality n.

One can generalize Algorithm 2.1 to come up with an estimator for PER A: let us sample numbers u_{ij} independently and at random from the standard Gaussian distribution in $\mathbb{R}$, cf. Section 2.1. Let us compute an $n \times m$ matrix $B = (b_{ij})$, where

$b_{ij} = u_{ij}\sqrt{a_{ij}}$. Finally, let $\alpha = \det(BB^*)$. Thus α is a non-negative number. Using an identity from linear algebra (see the Binet-Cauchy formula in Section 2 of [26]), we can write

$$\det(BB^*) = \sum_{|I|=n} (\det B_I)^2,$$

where the sum is taken over all subsets $I \subset \{1, \dots, m\}$ of the cardinality n and B_I is the submatrix consisting of the columns indexed by I. Since the expectation of every summand is the corresponding permanent per A_I (see Theorem 2.1.1), we conclude that α is an unbiased estimator. Let us introduce vectors $u_i \in \mathbb{R}^m$, $u_i = (u_{i1}, \dots, u_{im})$ for $i = 1, \dots, n$. Then α is a function of $u_1, \dots, u_n$ and $\alpha(u_1, \dots, u_n)$ satisfies the properties a)–c) above. Hence we get a randomized polynomial time algorithm that approximates α within a factor of $O(c^n)$. Note that n is the *smaller* dimension of the matrix A. As in Theorem 2.1.1, we have $c \approx 0.28$. Similarly, a complex estimator can be constructed, which gives $c \approx 0.56$. A quaternionic version with $c \approx 0.76$ is more complicated; it requires computation of a certain Pfaffian.

It appears that the condition b) can be replaced by a weaker condition:

b°) For any fixed $u_1, \dots, u_{i-1}, u_{i+1}, \dots, u_n$, the function $\alpha(u_1, \dots, u_n)$ is a *quadratic polynomial* (not necessarily homogeneous) in u_i.

One can construct some interesting estimators satisfying this weaker property.

Example. Approximating the hafnian. Let A be an $n \times n$ non-negative symmetric matrix. Suppose that n is even, $n = 2k$. The number

$$\text{haf } A = \frac{1}{2^k k!} \sum_{\sigma \in S_n} \prod_{i=1}^{k} a_{\sigma(2i-1), \sigma(2i)}$$

is called the *hafnian* of A, see Section 8.2 of [25]. Thus haf A is the sum of all monomials $a_{i_1 j_1} \cdots a_{i_k j_k}$, where $\{i_1, j_1\}, \dots, \{i_k, j_k\}$ constitutes a partition of the set $\{1, \dots, n\}$ into unordered pairs. For example, if A is the adjacency matrix of an undirected graph, haf A is the number of perfect matchings in the graph.

Let us sample $u_{ij} : 1 \leq i < j \leq n$ independently and at random from the standard Gaussian distribution in $\mathbb{R}$. Let us construct a skew-symmetric matrix $B = (b_{ij})$, where

$$b_{ij} = \begin{cases} u_{ij}\sqrt{a_{ij}} & \text{if } 1 \leq i < j \leq n, \\ -u_{ij}\sqrt{a_{ij}} & \text{if } 1 \leq j < i \leq n, \\ 0 & \text{if } i = j. \end{cases}$$

Let $\alpha = \det B$.

Let us introduce vectors $u_i \in \mathbb{R}^{n-i}$, $i = 1, \dots, n-1$ where $u_i = (u_{ii+1}, \dots, u_{in})$. Then α is a function of $u_1, \dots, u_{n-1}$. One can prove that α satisfies the properties a), b°), and c) (cf. Chapter 8 of [23]) and that with high probability α approximates haf A within a factor of $O(c^n)$, where $1 > c > 0$ is an absolute constant. Similarly, a complex estimator can be constructed and the author has a conjecture what a quaternionic version may look like.

Finally, an interesting question is what can we gain (or lose) by further relaxing b°) to

b*) For any fixed $u_1, \dots, u_{i-1}, u_{i+1}, \dots, u_n$, the function $\alpha(u_1, \dots, u_n)$ is a polynomial of a fixed (independent of n) degree.

These and related questions will be addressed elsewhere.

8. Applications of Mixed Discriminants to Counting

For a vector $x \in \mathbb{R}^n$, $x = (\xi_1, \dots, \xi_n)$, let us denote by $x \otimes x$ the $n \times n$ matrix whose (i,j)-th entry is $\xi_i \cdot \xi_j$. Thus $x \otimes x$ is a positive semidefinite matrix whose rank does not exceed 1.

Applications of mixed discriminants to problems of combinatorial counting are based on the following simple result.

(8.1) Lemma. *Let $u_1, \dots, u_n$ be vectors from $\mathbb{R}^n$. Then*

$$D(u_1 \otimes u_1, \dots, u_n \otimes u_n) = \Big(\det[u_1, \dots, u_n]\Big)^2,$$

the squared determinant of the matrix with the columns $u_1, \dots, u_n$.

Proof. Let $e_1, \dots, e_n$ be the standard orthonormal basis of $\mathbb{R}^n$. Let G be the matrix with the columns $u_1, \dots, u_n$. Then $u_i = Ge_i$, $u_i \otimes u_i = G(e_i \otimes e_i)G^*$ and from the definition of the mixed discriminant (see Section 1.2), we get

$$D(u_1 \otimes u_1, \dots, u_n \otimes u_n) = \frac{\partial^n}{\partial t_1 \dots \partial t_n} \det\Big(t_1 u_1 \otimes u_1 + \dots + t_n u_n \otimes u_n\Big)$$

$$= \frac{\partial^n}{\partial t_1 \dots \partial t_n} \det\Big(G(t_1 e_1 \otimes e_1 + \dots + t_n e_n \otimes e_n)G^*\Big)$$

$$= \det(GG^*) \frac{\partial^n}{\partial t_1 \dots \partial t_n} \det\Big(t_1 e_1 \otimes e_1 + \dots + t_n e_n \otimes e_n\Big) = (\det G)^2.$$

□

Suppose we are given a rectangular $n \times m$ matrix A with the columns $u_1, \dots, u_m$, which we interpret as vectors from $\mathbb{R}^n$. Suppose that for any subset $I \subset \{1, \dots, m\}$, $I = \{i_1, \dots, i_n\}$, the determinant of the submatrix $A_I = [u_{i_1}, \dots, u_{i_n}]$ is either 0, -1 or 1. Such an A represents a *unimodular matroid* on the set $\{1, \dots, m\}$ and a subset I with $\det A_I \neq 0$ is called a *base* of the matroid (see [29]).

Suppose now that the columns of A are colored with n different colors. The coloring induces a partition $\{1, \dots, m\} = J_1 \cup \dots \cup J_n$. We are interested in the number of bases that have precisely one index of each color. Let us define the positive semidefinite matrices $Q_1, \ldots, Q_n$ as follows:

$$Q_k = \sum_{i \in J_k} u_i \otimes u_i, \quad k = 1, \ldots, n.$$

Then the number of bases can be expressed as $D(Q_1, \ldots, Q_n)$. The mixed discriminant is linear in every argument; that is,

$$D(Q_1, \ldots, Q_{i-1}, \alpha Q'_i + \beta Q''_i, Q_{i+1} \ldots, Q_n)$$
$$= \alpha D(Q_1, \ldots, Q_{i-1}, Q'_i, Q_{i+1}, \ldots, Q_n) + \beta D(Q_1, \ldots, Q_{i-1}, Q''_i, Q_{i+1}, \ldots, Q_n)$$

(see, for example, [21]). Using the linearity and Lemma 8.1, we have

$$\begin{aligned} D(Q_1, \ldots, Q_n) &= \sum_{I=\{i_1, \ldots, i_n\}} D(u_{i_1} \otimes u_{i_1}, \ldots, u_{i_n} \otimes u_{i_n}) \\ &= \sum_{I=\{i_1, \ldots, i_n\}} \Big(\det[u_{i_1}, \ldots, u_{i_n}]\Big)^2, \end{aligned}$$

where the sums are taken over all n-subsets I, having precisely one element of each color, and the proof follows. R. Stanley obtained a similar formula which involves the mixed volume of zonotopes instead of the mixed discriminant [28].

Example. Trees in a graph. Suppose we have a connected graph G with n vertices and m edges. Suppose further that the edges of G are colored with $n-1$ different colors. We are interested in spanning trees T in G such that all edges of T have different colors. Let us number the vertices of G by $1, \ldots, n$ and the edges of G by $1, \ldots, m$. Let us make G an oriented graph by orienting its edges arbitrarily. We consider the truncated *incidence matrix* (with the last row removed) $A = (a_{ij})$ for $1 \leq i \leq n-1$ and $1 \leq j \leq m$ as an $(n-1) \times m$ matrix such that

$$a_{ij} = \begin{cases} 1 & \text{if } i \text{ is the beginning of } j \\ -1 & \text{if } i \text{ is the end of } j \\ 0 & \text{otherwise.} \end{cases}$$

The spanning trees of G are in a one-to-one correspondence with non-degenerate $(n-1) \times (n-1)$ submatrices of A and the determinant of a such a submatrix is either 1 or -1 (see, for example, Chapter 4 of [10]). Hence counting colored trees reduces to computing the mixed discriminant of some positive semidefinite matrices, computed from the incidence matrix of the graph.

Added in proofs: Combinatorial applications of mixed discriminants described in Section 8 are known, see Chapter 5 of R.B. Bapat and T.E.S. Raghavan, *Nonnegative Matrices and Applications*, Cambridge University Press, 1997.

Acknowledgment

I am grateful to Mark Jerrum and Martin Dyer for many useful conversations during the Schloß Dagstuhl meeting on approximation algorithms in August, 1997. It was their suggestion to explore a complex version (now Algorithm 2.2) of the original real algorithm (now Algorithm 2.1).

References

1. A. Aho, J. Hopcroft and J. Ullman, *The Design and Analysis of Computer Algorithms*, Addison-Wesley, Reading, 1974.
2. A.D. Aleksandrov, On the theory of mixed volumes of convex bodies, IV, Mixed discriminants and mixed volumes (in Russian), *Matematicheskii Sbornik, Novaya Seriya*, **3**(1938), 227–251.
3. E. Artin, *Geometric Algebra*, Interscience, Wiley, New York, 1988.
4. R.G. Bartle, *The Elements of Integration*, John Wiley & Sons, New York, 1966.
5. A.I. Barvinok, Two algorithmic results for the Traveling Salesman Problem, *Mathematics of Operations Research*, **21**(1996), 65–84.
6. A. I. Barvinok, Computing mixed discriminants, mixed volumes, and permanents, *Discrete & Computational Geometry*, **18**(1997), 205–207.
7. A.I. Barvinok, A simple polynomial time algorithm to approximate the permanent within a simply exponential factor, *preprint*, MSRI, No. 1997-031.
8. M. Dyer and M. Jerrum, *personal communication*, Schloß Dagstuhl Meeting on Approximation Algorithms, August 1998.
9. G.P. Egorychev, The solution of van der Waerden's problem for permanents, *Advances in Mathematics*, **42** (1981), 299–305.
10. V.A. Emelichev, M.M. Kovalev, and M.K. Kravtsov, *Polytopes, Graphs and Optimization*, Cambridge University Press, New York, 1984.
11. V.N. Faddeeva, *Computational Methods of Linear Algebra*, Dover, 1975.
12. A. Frieze and M. Jerrum, An analysis of a Monte Carlo algorithm for estimating the permanent, *Combinatorica*, **15** (1995), 67–83.
13. C.D. Godsil and I. Gutman, On the matching polynomial of a graph, in: *Algebraic Methods in Graph Theory, Vol. I, II, (Szeged, 1978)*, North-Holland, Amsterdam - New York, 1981, 241–249.
14. D.Yu. Grigoriev and M. Karpinski, The matching problem for bipartite graphs with polynomially bounded permanents is in NC, *Proceedings of the Twenty-Eight Annual IEEE Symposium on Foundations of Computer Science,* IEEE Computer Soc. Press, Washington D.C., 1987, 162–172.
15. M. Jerrum and A. Sinclair, Approximating the permanent, *SIAM Journal on Computing*, **18**(1989), 1149–1178.
16. M. Jerrum and A. Sinclair, The Markov chain Monte Carlo method: an approach to approximate counting and integration, in: *Approximation Algorithms for NP-hard Problems*, D.S. Hochbaum, ed., PWS Publishing, Boston, MA, 1996, 482–520.
17. M.R. Jerrum, L.G. Valiant and V.V. Vazirani, Random generation of combinatorial structures from a uniform distribution, *Theoretical Computer Science*, **43** (1986), 169–188.
18. M. Jerrum and U. Vazirani, A mildly exponential approximation algorithm for the permanent, *Algorithmica*, **16** (1996), 392–401.
19. N. Karmarkar, R. Karp, R. Lipton, L. Lovász and M. Luby, A Monte Carlo algorithm for estimating the permanent, *SIAM Journal on Computing*, **22**(1993), 284–293.

20. D.E. Knuth, *The Art of Computer Programming. Vol. 2: Seminumerical Algorithms*, second edition, Addison-Wesley, MA 1981.
21. K. Leichtweiß, Convexity and Differential Geometry, in: *Handbook of Convex Geometry*, ed. P.M. Gruber and J.M. Wills, vol. B, Chapter 4.1, North-Holland, 1993, 1045–1080.
22. N. Linial, A. Samorodnitsky and A. Wigderson, A deterministic strongly polynomial algorithm for matrix scaling and approximate permanents, *preprint.*
23. L. Lovász and M.D. Plummer, *Matching Theory*, North - Holland, Amsterdam - New York and Akadémiai Kiadó, Budapest, 1986.
24. L. Lovász and M. Simonovits, Random walks in a convex body and an improved volume algorithm, *Random Structures & Algorithms* **4**(1993), 359–412.
25. H. Minc, *Permanents*, Addison-Wesley, Reading, MA, 1978.
26. V.V. Prasolov, *Problems and Theorems in Linear Algebra*, American Mathematical Society, RI, 1994.
27. L.E. Rasmussen, Approximating the permanent: a simple approach, *Random Structures & Algorithms*, **5**(1994), 349–361.
28. R.P. Stanley, Two combinatorial applications of the Alexandrov-Fenchel inequalities, *Journal of Combinatorial Theory*, Ser. A, **17**(1981), 56–65.
29. N. White, Unimodular matroids, in: *Combinatorial Geometries*, Cambridge Univ. Press, Cambridge, 1987, 40–52.