10-14-2021

Genz and Mendell-Elston Estimation of the High-Dimensional Multivariate Normal Distribution

Lucy Blondell
The University of Texas Rio Grande Valley

Mark Kos
The University of Texas Rio Grande Valley

John Blangero
The University of Texas Rio Grande Valley

Harald H. H. Goring
The University of Texas Rio Grande Valley

Recommended Citation

Blondell, L.; Koz, M.Z.; Blangero, J.; Göring, H.H.H. Genz and Mendell-Elston Estimation of the High-Dimensional Multivariate Normal Distribution. Algorithms 2021, 14, 296. https://doi.org/10.3390/a14100296

algorithms

MDPI

Article

Genz and Mendell-Elston Estimation of the High-Dimensional Multivariate Normal Distribution

Lucy Blondell *, Mark Z. Kos, John Blangero and Harald H. H. Göring

Department of Human Genetics, South Texas Diabetes and Obesity Institute, University of Texas Rio Grande Valley, 3463 Magic Drive, San Antonio, TX 78229, USA; mark.kos@utrgv.edu (M.Z.K.); john.blangero@utrgv.edu (J.B.); harald.goring@utrgv.edu (H.H.H.G.)

* Correspondence: lucy.blondell@utrgv.edu

Abstract: Statistical analysis of multinomial data in complex datasets often requires estimation of the multivariate normal (MVN) distribution for models in which the dimensionality can easily reach 10–1000 and higher. Few algorithms for estimating the MVN distribution can offer robust and efficient performance over such a range of dimensions. We report a simulation-based comparison of two algorithms for the MVN that are widely used in statistical genetic applications. The venerable Mendell-Elston approximation is fast but execution time increases rapidly with the number of dimensions, estimates are generally biased, and an error bound is lacking. The correlation between variables significantly affects absolute error but not overall execution time. The Monte Carlo-based approach described by Genz returns unbiased and error-bounded estimates, but execution time is more sensitive to the correlation between variables. For ultra-high-dimensional problems, however, the Genz algorithm exhibits better scale characteristics and greater time-weighted efficiency of estimation.

Keywords: Genz algorithm; Mendell-Elston algorithm; multivariate normal distribution; Monte Carlo integration

Citation: Blondell, L.; Koz, M.Z.; Blangero, J.; Göring, H.H.H. Genz and Mendell-Elston Estimation of the High-Dimensional Multivariate Normal Distribution. *Algorithms* **2021**, *14*, 296. https://doi.org/10.3390/a14100296

Academic Editor: Tom Burr

Received: 5 August 2021
Accepted: 13 October 2021
Published: 14 October 2021

1. Introduction

In applied multivariate statistical analysis one is frequently faced with the problem of estimating the multivariate normal (MVN) distribution (or, equivalently, integrating the MVN density) not only for a range of correlation or covariance structures, but also for a number of dimensions (i.e., variables) that can span several orders of magnitude. In applications for which only one or a few instances of the distribution, and of low dimensionality ($n \lesssim 10$), must be estimated, conventional numerical methods based on, e.g., Newton-Cotes formulæ, Gaussian quadrature and orthogonal polynomials, or tetrachoric series, may offer satisfactory combinations of computational speed and estimation precision.

Increasingly, however, statistical analysis of large datasets requires many evaluations of very high-dimensional MVN distributions—often as an incidental part of some larger analysis—and places severe demands on the requisite speed and accuracy of numerical methods. We confront the need to estimate the high-dimensional MVN integral in statistical genetics, and particularly in genetic analyses of extended pedigrees (i.e., large, multigenerational collections of related individuals). A typical exercise is variance component analysis of a discrete trait (e.g., a qualitative or categorical measurement of some disease or other condition of interest) under a liability threshold model [1–3]. Maximum-likelihood estimation of the model parameters in such an application can easily require tens or hundreds of evaluations of the MVN distribution for which $n \approx 100$–1000 or greater [4–7], and situations in which $n \approx 10{,}000$ are not unrealistic.

In such problems the dimensionality of the model distribution is determined by the product of the total number of individuals in the pedigree(s) to be analyzed and the number of discrete phenotypes jointly analyzed [1,8]. For univariate traits studied in small pedigrees, such as sibships (sets of individuals born to the same parents) and nuclear families

(sibships and their parents), the dimensionality is typically small ($n \approx 2$–10), but analysis of multivariate phenotypes in large extended pedigrees routinely necessitates estimation of MVN distributions for which n can easily reach several thousand [2,3,7]. A single variance component-based linkage analysis of a univariate discrete phenotype in a set of extended pedigrees involves estimating these high-dimensional MVN distributions at hundreds of locations in the genome [3,9,10]. In these numerically-intensive applications, estimation of the MVN distribution represents the main computational bottleneck, and the performance of algorithms for estimation of the MVN distribution is of paramount importance.

Here we report the results of a simulation-based comparison of the performance of two algorithms for estimation of the high-dimensional MVN distribution, the widely-used Mendell-Elston (ME) approximation [1,8,11,12] and the Genz Monte Carlo (MC) procedure [13,14]. Each of these methods is well known, but previous studies have not investigated their properties for very large numbers of dimensions. Using conventional numerical desiderata of estimation accuracy, execution time, and computational efficiency, we examine the performance of these algorithms and identify aspects of the overall MVN context to which each method is particularly well suited.

In Section 2 we give a brief overview of techniques for estimating the MVN distribution. The ME and Genz MC algorithms are reviewed in Section 3. Procedures for exercising the algorithms and comparing their performance are described in Section 4, and results of the comparisons are presented in Section 5. In Section 6 we consider the interpretation and broader implications of our results for future applications.

2. Background

Numerical methods for estimation of the MVN distribution have a long and fascinating history of development and many interesting accounts from varied perspectives have been presented ([15–18], and references therein). Classical approaches to the problem have generally been variations on a few standard methods [19,20]. Often, some form of numerical quadrature is involved, in which an estimate of the integral is formed by accumulating a weighted sum of integrand values at a sequence of abcissæ covering the region of integration [21–24]. Tetrachoric series expansions [25,26] offer another approach to the problem, although these series may converge slowly, and in fact do not converge at all at some points in the correlation space for a given number of dimensions [27]. Other approaches have involved quadrature applied to an integral transformation of the tetrachoric series [19,28,29], and decomposition of the multidimensional probability into a product of conditional univariate probabilities [1,8,11,12,30–33].

In practice, the utility and applicability of any algorithm for estimating the MVN distribution is overwhelmingly constrained by the dimensionality of the problem. Fast and accurate algorithms have been described for evaluation of the univariate and multivariate normal distributions for 'small' numbers of dimensions; for the frequently encountered cases of $n = 1$ [34] and $n = 2$ [35–38], several algorithms are available that can in principle provide any desired accuracy. For the case $n > 2$, several algorithms (error-bounded and not) based on quadrature have been developed and their relative performance compared [17,21–24]. Monte Carlo approaches to the problem have also been developed that have desirable statistical properties and exhibit good scale properties with the number of dimensions [13,14,39,40].

As the number of dimensions reaches $n \approx 10$, many approaches to estimating the MVN distribution become impractical. Conventional series approximations and quadrature methods grow unwieldy, and the computational burden for algorithms using these methods rapidly becomes prohibitive [13,14,19,20,41]. However, methods of estimation based on reduction or transformation of the joint n-variate distribution to a series of (typically univariate) integrals continue to scale favorably with the number of dimensions.

3. Algorithms

We examined the performance of two algorithms that appear particularly well suited to estimation of the high-dimensional MVN distribution. The first of these is the Mendell-Elston (ME) procedure (Algorithm 1), a deterministic, non-error-bounded procedure that approximates the MVN distribution as a sequence of conditional univariate normal integrals [1,8,11,12]. The second algorithm is the elegant transformation and estimation procedure described by Genz [13,14] (Algorithm 2). In this approach the original n-variate distribution is transformed into an easily sampled $(n-1)$-dimensional hypercube and estimated by Monte Carlo methods (e.g., [42,43]).

Algorithm 1 Mendell-Elston Estimation of the MVN Distribution [12].

Estimate the standardized n-variate MVN distribution, having zero mean and correlation matrix $\mathbf{R}$, between vector-valued limits $\mathbf{s}$ and $\mathbf{t}$. The function $\phi(z)$ is the univariate normal density at z, and $\Phi(z)$ is the corresponding univariate normal distribution. See Hasstedt [12] for discussion of the approximation, extensions, and applications.

1. **input** n, $\mathbf{R}$, $\mathbf{s}$, $\mathbf{t}$
2. **initialize** $f = 1$
3. for $i = 1, 2, \ldots, n$

   (a) *[update the total probability]*

   $$p_i = \Phi(t_i) - \Phi(s_i)$$
   $$f \leftarrow f \cdot p_i$$
   if $(i = n)$ return f

   (b) *[peel variable i]*

   $$a_i = \frac{\phi(s_i) - \phi(t_i)}{\Phi(t_i) - \Phi(s_i)}$$
   $$V_i = 1 + \frac{s_i\,\phi(s_i) - t_i\,\phi(t_i)}{\Phi(t_i) - \Phi(s_i)} - a_i^2$$
   $$v_i^2 = 1 - V_i$$

   (c) *[condition the remaining variables]*

   for $j = i+1, \ldots, n$, $\quad k = j+1, \ldots, n$
   $$s'_j = \left(s_j - r_{ij}\,a_i\right) / \sqrt{1 - r_{ij}^2\,v_i^2}$$
   $$t'_j = \left(t_j - r_{ij}\,a_i\right) / \sqrt{1 - r_{ij}^2\,v_i^2}$$
   $$V'_j = V_j / \left(1 - r_{ij}^2\,v_i^2\right)$$
   $$v'^2_j = 1 - V'_j$$
   $$r'_{jk} = \left(r_{jk} - r_{ij}\,r_{ik}\,v_i^2\right) / \left(\sqrt{1 - r_{ij}^2\,v_i^2}\sqrt{1 - r_{ik}^2\,v_i^2}\right)$$

   [end loop over j,k]

[end loop over i]

The ME approximation is extremely fast, and broadly accurate over much of the parameter space [1,8,17,41]. The chief source of error in the approximation derives from the assumption that, at each stage of conditioning, the selected and unselected variables continue to distribute in approximately normal fashion [1]. This assumption is analytically true only for the initial stage(s) of selection and conditioning [17]; in subsequent stages the assumption is violated to greater or lesser degree and introduces error into the

approximation [31,33,44,45]. Consequently, the ME approximation is most accurate for small correlations and for selection in the tails of the distribution, thereby minimizing departures from normality following selection and conditioning. Conversely, the error in the ME approximation is greatest for larger correlations and selection closer to the mean [1].

Algorithm 2 Genz Monte Carlo Estimation of the MVN Distribution [13].

Estimate the m-variate MVN distribution having covariance matrix $\mathbf{\Sigma}$, between vector-valued limits $\mathbf{a}$ and $\mathbf{b}$, to an accuracy ϵ with probability $1 - \alpha$, or until the maximum number of integrand evaluations $N_{\max}$ is reached. The procedure returns the estimated probability F, the estimation error Δ, and the number of iterations N. The function $\Phi(x)$ is the univariate normal distribution at x, $\Phi^{-1}(x)$ is the corresponding inverse function; $u(\,)$ is a source of uniform random deviates on $(0, 1)$; and $Z_{\alpha/2}$ is the two-tailed Gaussian confidence factor corresponding to α. See Genz [13,14] for discussion, a worked example, and suggestions for optimizing algorithm performance.

1. **input** m, $\mathbf{\Sigma}$, $\mathbf{a}$, $\mathbf{b}$, ϵ, α, $N_{\max}$
2. compute the Cholesky decomposition $\mathbf{CC}'$ of $\mathbf{\Sigma}$
3. **initialize** $I = 0$, $V = 0$, $N = 0$, $d_1 = \Phi(a_1/c_{11})$, $e_1 = \Phi(b_1/c_{11})$, $f_1 = (e_1 - d_1)$
4. **repeat**
   (a) for $i = 1, 2, \ldots, m-1$
   $w_i \leftarrow u(\,)$
   (b) for $i = 2, 3, \ldots, m$
   $y_{i-1} = \Phi^{-1}[d_{i-1} + w_{i-1}(e_{i-1} - d_{i-1})]$
   $t_i = \sum_{j=1}^{i-1} c_{ij}\, y_j$
   $d_i = \Phi[(a_i - t_i)/c_{ii}]$
   $e_i = \Phi[(b_i - t_i)/c_{ii}]$
   $f_i = (e_i - d_i) f_{i-1}$
   (c) **update** $I \leftarrow I + f_m$, $V \leftarrow V + f_m^2$, $N \leftarrow N + 1$
   (d) $\Delta = Z_{\alpha/2}\sqrt{[(V/N - (I/N)^2]/N}$
   until $(\Delta < \epsilon)$ or $(N = N_{\max})$
5. $F = I/N$
6. **return** F, Δ, N

Despite taking somewhat different approaches to the problem of estimating the MVN distribution, these algorithms have some features in common. Most significantly, both algorithms reformulate the initial n-dimensional integral as a series of univariate integrals. This feature facilitates imposing an initial ordering of variables to minimize the potential loss of precision as the integral estimate is accumulated. In similar fashion, prioritizing variables appropriately can also help minimize error in the ME method introduced by violations of the assumptions underlying the method [17].

4. Algorithm Comparison

4.1. Program Implementation

Programs implementing the ME and MC approximations were written in ANSI C following published algorithms [12,13]. Implementation of the ME approximation follows the procedure described by Hasstedt [12] for likelihood evaluation of arbitrary mixtures of MVN densities and distributions. Although the algorithm in [12] is presented in the context of statistical genetics, it is a completely general formulation of the ME method and suitable for any application requiring estimation of the MVN distribution. Implementation of the MC approximation directly follows the algorithm presented by Genz [13].

To facilitate testing a simple driver program was written for each algorithm. The driver program accepts arguments defining the estimation problem (e.g., number of dimensions, correlations, limits of integration), and any algorithm-specific parameters (e.g., convergence criteria). The driver program then initializes the problem (i.e., generates the correlation matrix and limits of integration), calls the algorithm, records its execution time, and reports results. For the deterministic ME algorithm there are no essential user options; the only input quantities are those defining the MVN distribution and region of integration. The driver program for the Genz MC algorithm provides options for setting parameters unique to Monte Carlo estimation such as the (maximum) error in the estimate and the (maximum) allowed number of iterations (integrand evaluations) [13].

The actual software implementation of the estimation procedures and their respective driver programs is not critical; experiments with multiple independent implementations of these algorithms have shown consistent and reliable performance irrespective of programming language or style [2,3,7,10,46]. Attention to programming esoterica—e.g., selective use of alternative numerical techniques according to the region of integration, supplementing iterative estimation with functional approximations or table lookup methods, devolving the original integral as a sequence of conditional oligovariate (rather than univariate) problems—could conceivably yield modest improvements in execution times in some applications.

4.2. Test Problems

For validating and comparing the MC and ME algorithms it is important to have a source of independently determined values of the MVN distribution against which to compare the approximations returned by each algorithm. For many purposes it may be sufficient to refer to tables of the MVN distribution that have been generated for special cases of the correlation matrix [15,18,47–51]. Here, however, as in similar numerical studies [1,8,14,41], values of the MVN distribution were computed independently for correlation matrices defined by

$$\mathbf{R}_n = \mathbf{I}_n + \rho(\mathbf{J}_n - \mathbf{I}_n) \tag{1}$$

where n is the number of dimensions, $\mathbf{I}$ is the identity matrix, $\mathbf{J} = \mathbf{1}\mathbf{1}'$ is a matrix of ones, and ρ is a correlation coefficient. For $\mathbf{R}_n$ of this form, the n-variate MVN distribution at $\mathbf{b} = (b_1, \ldots, b_n)'$ can be reduced to the single integral

$$I_n(\mathbf{b}) = \int_{-\infty}^{+\infty} \phi(t) \prod_{i=1}^{n} \Phi\left(\frac{b_i + t\sqrt{\rho}}{\sqrt{1-\rho}}\right) dt \quad , \tag{2}$$

where $\phi(t)$ is the univariate normal density at t and $\Phi(t)$ is the corresponding univariate normal distribution [18,47,49,50]. This result involves only univariate normal functions and can be computed to desired accuracy using standard numerical methods (e.g., [43]).

4.3. Test Conditions

Two series of comparisons were conducted. In the first series, algorithms were compared using correlation matrices $\mathbf{R}_n$ with $\rho \in \{0.1, 0.3, 0.5, 0.9\}$ and $n = 3(1)10$ (i.e., n from 3 to 10 by 1), $n = 10(10)100$, and $n = 100(100)1000$. The lower and upper limits of integration, respectively, were $a_i = -\infty$ and $b_i = 0$, $i = 1, \ldots, n$.

In the second series of comparisons, correlation matrices $\mathbf{R}_n$ were generated with values of ρ drawn randomly from the uniform distribution $U(0, 1)$ [52,53]; lower limits of integration remained fixed at $a_i = -\infty$, but upper limits b_i were chosen randomly from the uniform distribution $U(0, \sqrt{n})$.

For the Genz MC algorithm an initial estimate was generated using $N_0 = 100$ iterations (the actual value of N_0 was not critical); then, if necessary, iterations were continued (using $N_{k+1} = \frac{3}{2}N_k$) until the requested estimation accuracy ϵ was achieved [13,14]. Under the usual assumption that independent Monte Carlo estimates distribute normally about the

true integral value I, the $1-\alpha$ confidence interval for I is $\widetilde{I} \pm Z_{\alpha/2}\,\sigma_{\widetilde{I}}/\sqrt{n}$, where $\widetilde{I}$ is the estimated value, $\sigma_{\widetilde{I}}/\sqrt{n}$ is the standard error of $\widetilde{I}$, $Z_{\alpha/2}$ is the Monte Carlo confidence factor for the standard error, and α is the Type I error probability. Therefore, to achieve an error less than ϵ with probability $1-\alpha$, the algorithm samples the integral until $Z_{\alpha/2}\,\sigma_{\widetilde{I}}/\sqrt{n} < \epsilon$. For all results reported here we took $\alpha = 0.01$, corresponding to $Z_{\alpha/2} \approx 2.5758$.

4.4. Test Comparisons

Three aspects of algorithm performance were compared: the error in the estimate, the computation time required to generate the estimate, and the relative efficiency of estimation. One can invent many additional interesting and contextually relevant comparisons examining various aspects of estimation quality and algorithm performance, but the criteria used here have been applied in other studies (e.g., [39]), are simple to quantify, broadly relevant, and effective for delineating areas of the MVN problem space in which each method performs more or less optimally.

The estimation error is the difference between the estimate returned by the algorithm and the independently computed expectation. The computation time is the execution time required for the algorithm to return an estimate; for the MC procedure this quantity includes the (comparatively trivial) time required to obtain the Cholesky decomposition of the correlation matrix. The relative efficiency is the time-weighted ratio of the variance in each estimate (see, e.g., [39]). Thus, if t_{MC} and t_{ME}, respectively, denote the execution times of the MC and ME algorithms, and σ^2_{MC} and σ^2_{ME} the corresponding mean squared errors in the MC and ME estimates, then the relative efficiency is defined as $\theta = (t_{ME}\,\sigma^2_{ME})/(t_{MC}\,\sigma^2_{MC})$, i.e., the product of the relative mean-squared error $\sigma^2_{ME}/\sigma^2_{MC}$ and the relative execution time t_{ME}/t_{MC}. The measure is somewhat *ad hoc*, and in practical applications the choice of algorithm should ultimately be informed by pragmatic considerations but—*ceteris paribus*—values $\theta \gg 1$ tend to favor the Genz MC algorithm, and values $\theta \ll 1$ tend to favor the ME algorithm.

4.5. Computing Platforms

Numerical methods are of little use if they are ill-suited to the hardware available to the user. Both the ME and Genz MC algorithms involve the manipulation of large, nonsparse matrices, and the MC method also makes heavy use of random number generation, so there seemed no compelling reason *a priori* to expect these algorithms to exhibit similar scale characteristics with respect to computing resources. Algorithm comparisons were therefore conducted on a variety of computers having wildly different configurations of CPU, clock frequency, installed RAM, and hard drive capacity, including an intrepid Intel 386/387 system (25 MHz, 5 MB RAM), a Sun SPARCstation-5 workstation (160 MHz, 1 GB RAM), a Sun SPARCstation-10 server (50 MHz, 10 GB RAM), a Mac G4 PowerPC (1.5 GHz, 2 GB RAM), and a MacBook Pro with Intel Core i7 (2.5 GHz, 16 GB RAM). As expected, clock frequency was found to be the primary factor determining overall execution speed, but both algorithms performed robustly and proved entirely practical for use even with modest hardware. We did not, however, further investigate the effect of computer resources on algorithm performance, and all results reported below are independent of any specific test platform.

5. Results

5.1. Error

The errors in the estimates returned by each method are shown in Figure 1 for a single 'replication', i.e., an application of each algorithm to return a single (convergent) estimate. The figure illustrates the qualitatively different behavior of the two estimation procedures—the deterministic approximation returned by the ME algorithm, and the stochastic estimate returned by the Genz MC algorithm.

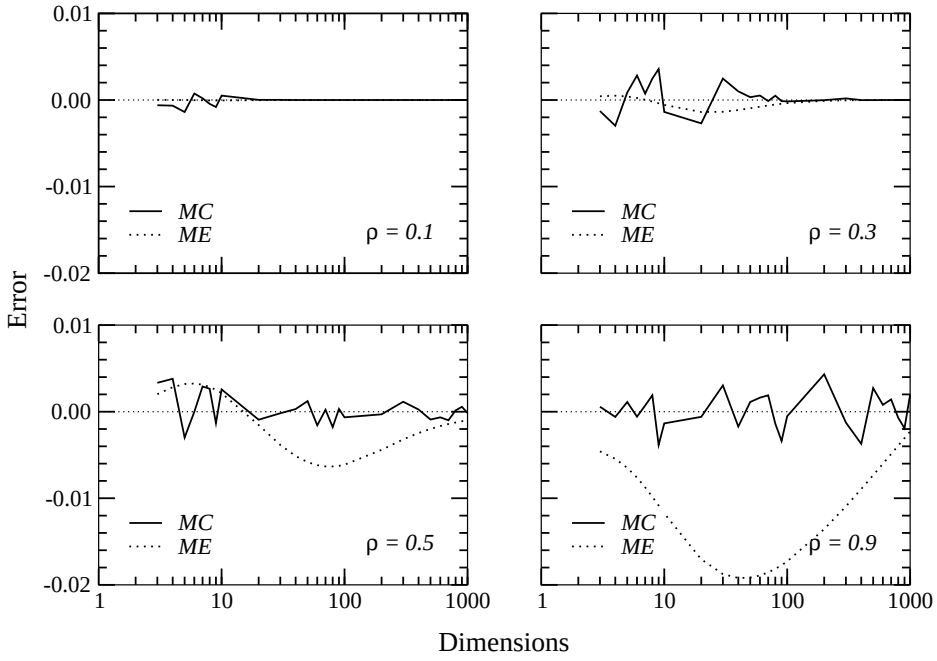

Figure 1. Estimation error in Genz Monte Carlo (MC) and Mendell-Elston (ME) approximations. (MC only: single replication; requested accuracy $\epsilon = 0.01$.)

Estimates from the MC algorithm are well within the requested maximum error for all values of the correlation coefficient and throughout the range of dimensions considered. Errors are unbiased as well; there is no indication of systematic under- or over-estimation with either correlation or number of dimensions.

In contrast, the error in the estimate returned by the ME method, though not generally excessive, is strongly systematic. For small correlations, or for moderate correlations and small numbers of dimensions, the error is comparable in magnitude to that from MC estimation but is consistently biased. For $\rho \gtrsim 0.3$, the error begins to exceed that of the corresponding MC estimate, and the desired distribution can be significantly under- or overestimated even for a small number of dimensions. This pattern of error in the ME approximation reflects the underlying assumption of multivariate normality of both the marginal and conditional distributions following variable selection [1,8,17]. The assumption is viable for small correlations, and for integrals of low dimensionality (requiring fewer iterations of selection and conditioning); errors are quickly compounded and the approximation deteriorates as the assumption becomes increasingly implausible.

Although bias in the estimates returned by the ME method is strongly dependent on the correlation among the variables, this feature should not discourage use of the algorithm. For example, estimation bias would not be expected to prejudice likelihood-based model optimization and estimation of model parameters, which are determined by the *location* of likelihood extrema. However, estimation bias could conceivably vitiate likelihood-ratio tests involving functions of the actual likelihood values. The latter may become of particular concern in applications that accumulate and compare likelihoods over a collection of independent data under varying model parameterizations.

5.2. Mean Execution Time

Relative mean execution time, t_{ME} and t_{MC} for the ME and MC algorithms respectively, is summarized in Figure 2 for 100 replications of each algorithm. As absolute execution times for a given application can vary by several orders of magnitude depending on com-

puting resources, the figure presents the ratio t_{ME}/t_{MC} which was found to be effectively independent of computing platform.

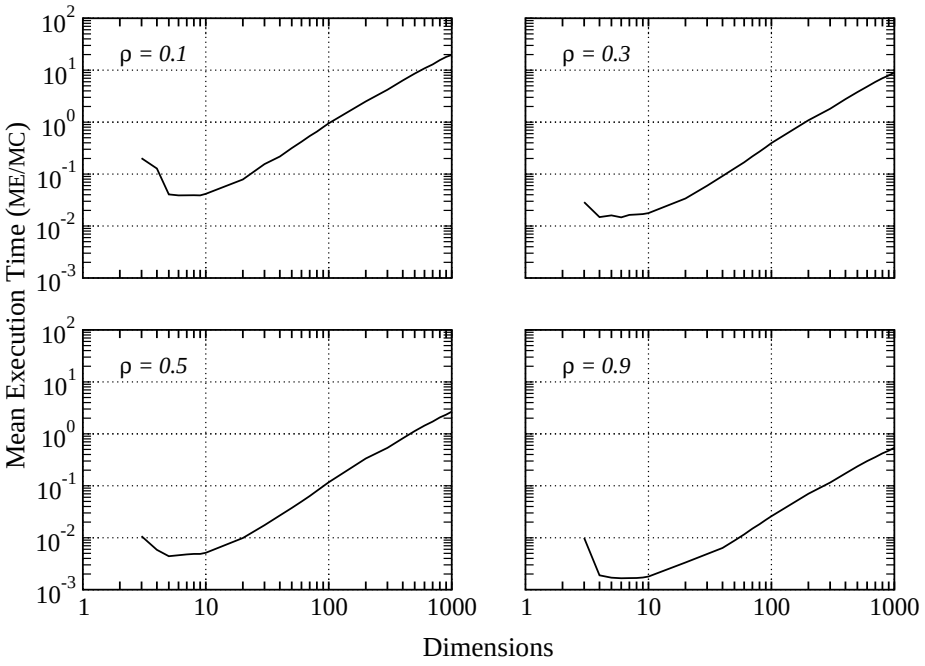

Figure 2. Relative mean execution time (t_{ME}/t_{MC}) of Genz Monte Carlo (MC) and Mendell-Elston (ME) algorithms. (MC only: mean of 100 replications; requested accuracy $\epsilon = 0.01$.)

For estimation of the MVN in moderately few dimensions ($n \lesssim 30$) the ME approximation is exceptionally fast. The mean execution time of the MC method can be markedly greater—e.g., at $n \approx 10$ about 10-fold slower for $\rho = 0.1$ and 1000-fold slower for $\rho = 0.9$. For small correlations the execution time of the MC method becomes comparable with that of the ME method for $n \approx 100$. For the largest numbers of dimensions considered, the Monte Carlo method can be substantially faster—nearly 10-fold when $\rho = 0.3$ and nearly 20-fold when $\rho = 0.1$.

The scale properties of mean execution time for the ME and MC algorithms with respect to correlation and number of dimensions may be important considerations for specific applications. The ME method exhibits virtually no variation in execution time with the strength of the correlation, which may be an attractive feature in applications for which correlations are highly variable and the dimensionality of the problem does not vary greatly. For the MC method, execution time increases approximately 10–fold as the correlation increases from $\rho = 0.1$ to $\rho = 0.9$, but is approximately constant with respect to the number of dimensions. This behavior would be desirable in applications for which correlations tend to be small but the number of dimensions varies considerably.

5.3. Relative Performance

In view of the statistical virtues of the MC estimate but the favorable execution times for the ME approximation, it is instructive to compare the algorithms in terms of a metric incorporating both of these aspects of performance. For this purpose we use the time- and error-weighted ratio used described by Deák [39], and compare the performance of the algorithms for randomly chosen correlations and regions of integration (see Section 4.3). As applied here, values of this ratio greater than one tend to favor the Genz MC method, and values less than one tend to favor the ME method.

The relative mean execution times, mean squared errors, and mean time-weighted efficiencies of the MC and ME methods are summarized in Figure 3. Although ME estimates can be markedly faster to compute—e.g., ∼100-fold faster for $n \approx 100$ and ∼10-fold faster

for $n \approx 1000$, *in these replications*)—the mean squared error of the MC estimates is consistently 10–100-fold smaller, and on this basis alone is the statistically preferable procedure. Measured by their time-weighted relative efficiency, however, the disparity in performance is less extreme; the ME algorithm is comparatively efficient for $n \lesssim 100$ dimensions, beyond which the MC algorithm becomes the more efficient approach.

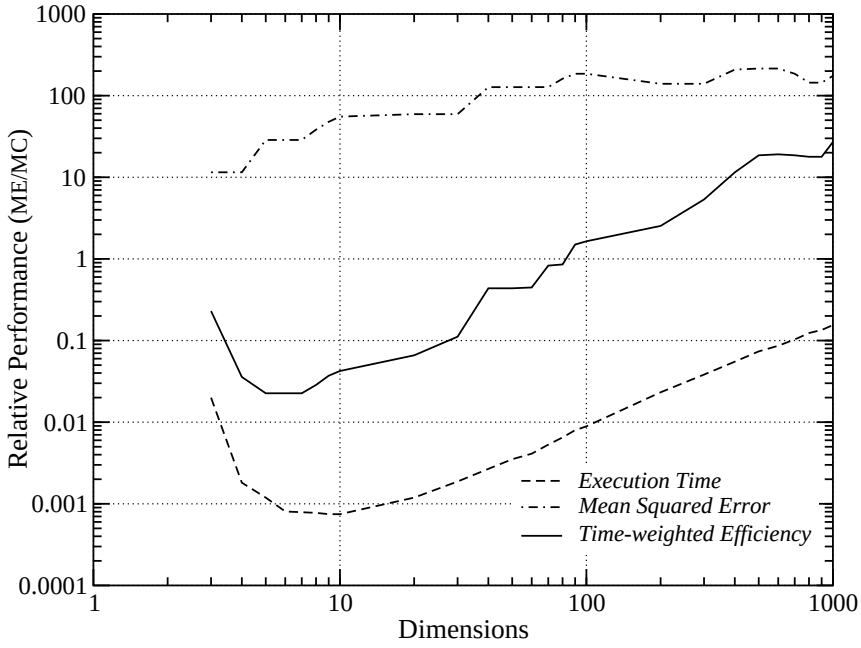

Figure 3. Relative performance of Genz Monte Carlo (MC) and Mendell-Elston (ME) algorithms: ratios of execution time, mean squared error, and time-weighted efficiency. (MC only: mean of 100 replications; requested accuracy $\epsilon = 0.01$.)

6. Discussion

Statistical methodology for the analysis of large datasets is demanding increasingly efficient estimation of the MVN distribution for ever larger numbers of dimensions. In statistical genetics, for example, variance component models for the analysis of continuous and discrete multivariate data in large, extended pedigrees routinely require estimation of the MVN distribution for numbers of dimensions ranging from a few tens to a few tens of thousands. Such applications reflexively (and understandably) place a premium on the sheer speed of execution of numerical methods, and statistical niceties such as estimation bias and error boundedness—critical to hypothesis testing and robust inference—often become secondary considerations.

We investigated two algorithms for estimating the high-dimensional MVN distribution. The ME algorithm is a fast, deterministic, non-error-bounded procedure, and the Genz MC algorithm is a Monte Carlo approximation specifically tailored to estimation of the MVN. These algorithms are of comparable complexity, but they also exhibit important differences in their performance with respect to the number of dimensions and the correlations between variables. We find that the ME algorithm, although extremely fast, may ultimately prove unsatisfactory if an error-bounded estimate is required, or (at least) *some* estimate of the error in the approximation is desired. The Genz MC algorithm, despite taking a Monte Carlo approach, proved to be sufficiently fast to be a practical alternative to the ME algorithm. Under certain conditions the MC method is competitive with, and can even outperform, the ME method. The MC procedure also returns unbiased estimates of desired precision, and is clearly preferable on purely statistical grounds. The MC method has excellent scale characteristics with respect to the number of dimensions, and greater overall estimation efficiency for high-dimensional problems; the procedure is somewhat more sensitive to the

correlation between variables, but this is not expected to be a significant concern unless the variables are known to be (consistently) strongly correlated.

For our purposes it has been sufficient to implement the Genz MC algorithm without incorporating specialized sampling techniques to accelerate convergence. In fact, as was pointed out by Genz [13], transformation of the MVN probability into the unit hypercube makes it possible for simple Monte Carlo integration to be surprisingly efficient. We expect, however, that our results are mildly conservative, i.e., underestimate the efficiency of the Genz MC method relative to the ME approximation. In intensive applications it may be advantageous to implement the Genz MC algorithm using a more sophisticated sampling strategy, e.g., non-uniform 'random' sampling [54], importance sampling [55,56], or subregion (stratified) adaptive sampling [13,57]. These sampling designs vary in their approach, applicability to a given problem, and computational overhead, but their common objective is to estimate the integral as efficiently as possible for a given amount of sampling effort. (For discussion of these and other variance reduction techniques in Monte Carlo integration, see [42,43].)

Finally, in choosing between these or other procedures for estimating the MVN distribution, it is helpful to observe a pragmatic distinction between applications that are deterministic and those that are genuinely stochastic in nature. The computational merits of fast execution time, accuracy, and precision may be advantageous for the analysis of well-behaved problems of a deterministic nature, yet be comparatively inessential for inherently statistical investigations. In many applications, some sacrifice in the speed of the algorithm (but not, as Figure 1 reveals, in the accuracy of estimation) could surely be tolerated in exchange for desirable statistical properties that promote robust inference [58]. These properties include unbiased estimation of the likelihood, an estimate of error instead of fixed error bounds (or no error bound at all), the ability to combine independent estimates into a variance-weighted mean, favorable scale properties with respect to the number of dimensions and the correlation between variables, and potentially increased robusticity to poorly-conditioned covariance matrices [20,42]. For many practical problems requiring the high-dimensional MVN distribution, the Genz MC algorithm clearly has much to recommend it.

Author Contributions: Conceptualization, L.B.; Data Curation, L.B.; Formal Analysis, L.B.; Funding Acquisition, H.H.H.G. and J.B.; Investigation, L.B.; Methodology, L.B.; Project Administration, H.H.H.G. and J.B.; Resources, J.B. and H.H.H.G.; Software, L.B.; Supervision, H.H.H.G. and J.B.; Validation, L.B.; Visualization, L.B.; Writing—Original Draft Preparation, L.B.; Writing—Review & Editing, L.B., M.Z.K. and H.H.H.G. All authors have read and agreed to the published version of the manuscript.

Funding: This research was supported in part by National Institutes of Health DK099051 (to H.H.H.G.) and MH059490 (to J.B.), a grant from the Valley Baptist Foundation (Project THRIVE), and conducted in part in facilities constructed under the support of NIH grant 1C06RR020547.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Rice, J.; Reich, T.; Cloninger, C.R.; Wette, R. An approximation to the multivariate normal integral: Its application to multifactorial qualitative traits. *Biometrics* **1979**, *35*, 451–459. [CrossRef]
2. Williams, J.T.; Eerdewegh, P.V.; Almasy, L.; Blangero, J. Joint multipoint linkage analysis of multivariate qualitative and quantitative traits. I. Likelihood formulation and simulation results. *Am. J. Hum. Genet.* **1999**, *65*, 1134–1147. [CrossRef]
3. Williams, J.T.; Begleiter, H.; Porjesz, B.; Edenberg, H.J.; Foroud, T.; Reich, T.; Goate, A.; Eerdewegh, P.V.; Almasy, L.; Blangero, J. Joint multipoint linkage analysis of multivariate qualitative and quantitative traits. II. Alcoholism and event-related potentials. *Am. J. Hum. Genet.* **1999**, *65*, 1148–1160. [CrossRef] [PubMed]

4. Falconer, D.S. The inheritance of liability to certain diseases, estimated from the incidence among relatives. *Ann. Hum. Genet.* **1965**, *29*, 51–76. [CrossRef]
5. Falconer, D.S. The inheritance of liability to diseases with variable age of onset, with particular reference to diabetes mellitus. *Ann. Hum. Genet.* **1967**, *31*, 1–20. [CrossRef] [PubMed]
6. Curnow, R.N.; Smith, C. Multifactorial models for familial diseases in man. *J. R Stat. Soc. A* **1975**, *138*, 131–169. [CrossRef]
7. Williams, J.T.; Blangero, J. Power of variance component analysis—II. Discrete traits. *Ann. Hum. Genet.* **2004**, *68*, 620–632. [CrossRef] [PubMed]
8. Mendell, N.R.; Elston, R.C. Multifactorial qualitative traits: Genetic analysis and prediction of recurrence risks. *Biometrics* **1974**, *30*, 41–57. [CrossRef]
9. Duggirala, R.; Williams, J.T.; Williams-Blangero, S.; Blangero, J. A variance component approach to dichotomous trait linkage analysis using a threshold model. *Genet. Epidemiol.* **1997**, *14*, 987–992. [CrossRef]
10. Williams, J.T.; Blangero, J. Efficient Monte Carlo evaluation of the multivariate normal integral. *Genet. Epidemiol.* **1998**, *15*, 540–541.
11. Mendell, N.R. Some Methods for Genetically Analyzing Human Qualitative Multifactorial Traits. Ph.D. Thesis, The University of North Carolina at Chapel Hill, Chapel Hill, NC, USA, 1972.
12. Hasstedt, S.J. Variance components/major locus likelihood approximation for quantitative, polychotomous, and multivariate data. *Genet. Epidemiol.* **1993**, *10*, 145–158. [CrossRef] [PubMed]
13. Genz, A. Numerical computation of multivariate normal probabilities. *J. Comp. Graph. Stat.* **1992**, *1*, 141–149.
14. Genz, A. Comparison of methods for the computation of multivariate normal probabilities. *Comput. Sci. Stat.* **1993**, *25*, 400–405.
15. Gupta, S.S. Probability integrals of multivariate normal and multivariate *t*. *Ann. Math. Stat.* **1963**, *34*, 792–828. [CrossRef]
16. Gupta, S.S. Bibliography on the multivariate normal integrals and related topics. *Ann. Math. Stat.* **1963**, *34*, 829–838. [CrossRef]
17. Eerdewegh, P.V. Statistical Selection in Multivariate Systems with Applications in Quantitative Genetics. Ph.D. Thesis, Washington University: St. Louis, MO, USA, 1982.
18. Tong, Y.L. *The Multivariate Normal Distribution*; Springer: New York, NY, USA, 1990.
19. Dutt, J.E. A representation of multivariate normal probability integrals by integral transforms. *Biometrika* **1973**, *60*, 637–645. [CrossRef]
20. Ducrocq, V.; Colleau, J.J. Interest in quantitative genetics of Dutt's and Deak's methods for numerical computation of multivariate normal probability integrals. *Génét. Sél. Evol.* **1986**, *18*, 447–474. [CrossRef]
21. Milton, R.C. Computer evaluation of the multivariate normal integral. *Technometrics* **1972**, *14*, 881–889. [CrossRef]
22. Bohrer, R.; Schervish, M.J. An error-bounded algorithm for normal probabilities of rectangular regions. *Technometrics* **1981**, *23*, 297–300. [CrossRef]
23. Schervish, M.J. Algorithm AS 195: Multivariate normal probabilities with error bound. *Appl. Stat.* **1984**, *33*, 81–94. [CrossRef]
24. Baigorri, A.R.; Eerdewegh, P.V.; Reich, T. *Error Bounded Integration of Multivariate Normal Densities over Rectangular Regions*; Technical Report; Department of Psychiatry, Washington University School of Medicine, St. Louis, MO, USA, 1986.
25. Pearson, K. III. Mathematical contributions to the theory of evolution. VIII. On the inheritance of characters not capable of exact quantitative measurement. *Philos. Trans. R Soc. Lond. A* **1901**, *195*, 79–148.
26. Kendall, M.G. Proof of relations connected with the tetrachoric series and its generalization. *Biometrika* **1941**, *32*, 196–198. [CrossRef]
27. Harris, B.; Soms, A.P. The use of the tetrachoric series for evaluating multivariate normal probabilities. *J. Multivar. Anal.* **1980**, *10*, 252–267. [CrossRef]
28. Dutt, J.E. On computing the probability integral of a general multivariate *t*. *Biometrika* **1975**, *62*, 201–205. [CrossRef]
29. Dutt, J.E.; Soms, A.P. An integral representation technique for calculating general multivariate probabilities with an application to multivariate χ^2. *Comm. Stat. Theory Meth.* **1976**, *A5*, 377–388.
30. Pearson, K. I. Mathematical contributions to the theory of evolution. XI. On the influence of natural selection on the variability and correlation of organs. *Philos. Trans. R Soc. Lond. A* **1903**, *200*, 1–66.
31. Soper, H.E. *Frequency Arrays*; Cambridge University Press: London, UK, 1922.
32. Aitken, A.C. Note on selection from a multivariate normal population. *Proc. Edinb. Math. Soc. Bull.* **1934**, *4*, 106–110. [CrossRef]
33. Lawley, D.N. A note on Karl Pearson's selection formulæ. *Proc. R. Soc. Edinb.* **1943**, *62*, 28–30. [CrossRef]
34. Hill, G.W. Algorithm 465: Student's *t* frequency [S14]. *Comm. ACM* **1973**, *16*, 690. [CrossRef]
35. Owen, D.B. Tables for computing bivariate normal probabilities. *Ann. Math. Stat.* **1956**, *27*, 1075–1090. [CrossRef]
36. Bender, H.A. Bivariate distribution. *Bull. Am. Math. Soc.* **1955**, *61*, 561–562.
37. Donnelly, T.G. Algorithm 462: Bivariate Normal Distribution [S15]. *Comm. ACM* **1973**, *16*, 638. [CrossRef]
38. Lowerre, J.M. An integral of the bivariate normal and an application. *Am. Stat.* **1983**, *37*, 235–236.
39. Deák, I. Three digit accurate multiple normal probabilities. *Numer. Math.* **1980**, *35*, 369–380. [CrossRef]
40. Deák, I. Computing probabilities of rectangles in case of multinormal distribution. *J. Stat. Comput. Simul.* **1986**, *26*, 101–114. [CrossRef]
41. Joe, H. Approximations to multivariate normal rectangle probabilities based on conditional expectations. *J. Am. Stat. Assoc.* **1995**, *90*, 957–964. [CrossRef]
42. Lepage, G.P. A new algorithm for adaptive multidimensional integration. *J. Comput. Phys.* **1978**, *27*, 192–203. [CrossRef]

43. Press, W.H.; Teukolsky, S.A.; Vetterling, W.T.; Flannery, B.P. *Numerical Recipes in C: The Art of Scientific Computing*, 2nd ed.; Cambridge University Press: Cambridge, UK, 1992.
44. Birnbaum, Z.W. Effect of linear truncation on a multinormal population. *Ann. Math. Stat.* **1950**, *21*, 272–279. [CrossRef]
45. Birnbaum, Z.W.; Paulson, E.; Andrews, F.C. On the effect of selection performed on some coordinates of a multi-dimensional population. *Psychometrika* **1950**, *15*, 191–204. [CrossRef]
46. L Almasy, J.B. Multipoint quantitative-trait linkage analysis in general pedigrees. *Am. J. Hum. Genet.* **1998**, *62*, 1198–1211. [CrossRef]
47. Curnow, R.N.; Dunnett, C.W. The numerical evaluation of certain multivariate normal integrals. *Ann. Math. Stat.* **1962**, *33*, 571–579. [CrossRef]
48. Kendall, M.G.; Stuart, A. *The Advanced Theory of Statistics. Volume 1. Distribution Theory*, 3rd ed.; Hafner: New York, NY, USA, 1969.
49. Curnow, R.N. The multifactorial model for the inheritance of liability to disease and its implications for relatives at risk. *Biometrics* **1972**, *28*, 931–946. [CrossRef] [PubMed]
50. Johnson, N.L.; Kotz, S. *Distributions in Statistics: Continuous Multivariate Distributions*, 2nd ed.; John Wiley & Sons: New York, NY, USA, 1972; Volume 4.
51. Six, F.B. Representations of multivariate normal distributions with special correlation structures. *Commun. Stat. Theory Meth.* **1981**, *10*, 1285–1295. [CrossRef]
52. Bendel, R.B.; Mickey, M.R. Population correlation matrices for sampling experiments. *Commun. Statist. Simul. Comput.* **1978**, *B7*, 163–182. [CrossRef]
53. Marsaglia, G.; Olkin, I. Generating correlation matrices. *SIAM J. Sci. Stat. Comput.* **1984**, *5*, 470–475. [CrossRef]
54. Bratley, P.; Fox, B.L. Algorithm 659: Implementing Sobol's quasirandom sequence generator. *ACM Trans. Math. Softw.* **1988**, *14*, 88–100. [CrossRef]
55. Hastings, W.K. Monte Carlo sampling methods using Markov chains and their applications. *Biometrika* **1970**, *57*, 97–109. [CrossRef]
56. Pandey, M.; Sarkar, A. Comparison of a simple approximation for multinormal integration with an importance sampling-based simulation method. *Probabilistic Eng. Mech.* **2002**, *17*, 215–218. [CrossRef]
57. Berntsen, J.; Espelid, T.O.; Genz, A. Algorithm 698: DCUHRE: An adaptive multidimensional integration routine for a vector of integrals. *ACM Trans. Math. Softw.* **1991**, *17*, 452–456. [CrossRef]
58. Zeng, Z. Precision mapping of quantitative trait loci. *Genetics* **1994**, *136*, 1457–1468. [CrossRef]