Computational Statistics and Data Analysis 55 (2011) 1521–1529

Contents lists available at ScienceDirect

Computational Statistics and Data Analysis

journal homepage: www.elsevier.com/locate/csda

Fast computation of high-dimensional multivariate normal probabilities☆

Ioannis Phinikettos*, Axel Gandy

Department of Mathematics, Imperial College London, London, SW7 2AZ, UK

ARTICLE INFO

Article history:
Received 17 December 2009
Received in revised form 28 September 2010
Accepted 1 October 2010
Available online 20 October 2010

Keywords:
Multivariate normal distribution
Monte Carlo methods
Singular values

ABSTRACT

A new efficient method is proposed to compute multivariate normal probabilities over rectangles in high dimensions. The method exploits four variance reduction techniques: conditional Monte Carlo, importance sampling, splitting and control variates. Simulation results are presented that evaluate the performance of the new proposed method. The new method is designed for computing small exceedance probabilities.

1. Introduction

In this article we present a new method for computing the probability that a multivariate normal random variable takes values in a rectangle $(a, b) = (a_1, b_1) \times \cdots \times (a_n, b_n)$, i.e. we are interested in

$$\Phi(a, b, \Sigma, \mu) = \int_{a_1}^{b_1} \cdots \int_{a_n}^{b_n} \frac{1}{|\Sigma|^{1/2}(2\pi)^{n/2}} e^{-\frac{1}{2}(x-\mu)'\Sigma^{-1}(x-\mu)} dx,$$

where $\mu = (\mu_1, \ldots, \mu_n)' \in \mathbb{R}^n$ and Σ is an $n \times n$ symmetric positive definite matrix.

Multivariate normal probabilities are a fundamental tool in many statistical applications. As a consequence, many different approaches have been proposed for computing these probabilities; see e.g. Deak (1980), Schervish (1984), Wang and Kennedy (1992), Genz (1992), Genz (1993), Vijverberg (1997), Somerville (1998), Szantai (2000), Miwa et al. (2003) and Sandor and Andras (2004). However, none of these methods was explicitly constructed for high-dimensional normal distributions.

High-dimensional multivariate normal probabilities are, for example, needed for the computation of p-values of certain model checks concerned with the detection of misspecified covariates in generalised linear models (Lin et al., 2002). The dimension in this case is the number of available observations. Statistical applications also include multiple comparisons (Dunnett, 1955), the multivariate probit model (Ochi and Prentice, 1984) and the multivariate ordinal response model (Anderson and Pemberton, 1985).

We will introduce a new method that exploits the diagonalisation of the covariance matrix. The diagonalisation imposes a priority ordering on the integration variables. Based on this, we apply various variance reduction techniques. These include conditional Monte Carlo, importance sampling, control variates and splitting. To the best of our knowledge, it is the first time that splitting is used to evaluate multivariate normal probabilities. The other three are standard techniques in this area.

☆ R-code implementing the suggested method is given in Appendix.

* Corresponding author.
E-mail addresses: ioannis.phinikettos@imperial.ac.uk (I. Phinikettos), a.gandy@imperial.ac.uk (A. Gandy).

0167-9473/$ – see front matter © 2010 Elsevier B.V. All rights reserved.
doi:10.1016/j.csda.2010.10.005

Genz (1992) and Genz and Bretz (2002) proposed a method for computing multivariate normal probabilities. A detailed description of this method can be found in Genz and Bretz (2009). It is, for example, implemented in R (R Development Core Team, 2010) and in Matlab. Our simulations show that our proposed algorithm performs better than Genz's method in certain situations.

The paper is organised as follows. Section 2 contains the description of the algorithm. In Section 3 we compare our method to previous approaches. In Section 4 we report simulation results. Concluding remarks are contained in Section 5.

2. Algorithm

The algorithm begins with a transformation of the integration region. Let $x = UDz$, where UD^2U' is the diagonalisation of the covariance matrix Σ. Here $D = \mathrm{diag}(d_1, \ldots, d_n)$ denotes the diagonal matrix of the positive square roots of the eigenvalues of the covariance matrix Σ in decreasing order and the columns of U are the corresponding orthonormal eigenvectors. Since $x'\Sigma^{-1}x = z'z$ and $\mathrm{d}x = |UD|\mathrm{d}z = |\Sigma|^{1/2}\mathrm{d}z$, we get

$$p \equiv \Phi(a, b, \Sigma, 0) = \frac{1}{(2\pi)^{n/2}} \int I(UDz \in A)\mathrm{e}^{-\frac{1}{2}z'z}\mathrm{d}z, \tag{1}$$

where $A = (a, b)$ and I denotes the indicator function. Without loss of generality, we assumed that $\mu = 0$.

Given this form, one can sample independent $Z^{(1)}, \ldots, Z^{(N)} \sim N_n(0, I)$ to form the crude Monte Carlo (MC) estimator

$$\hat{p}_{MC} = \frac{1}{N}\sum_{i=1}^{N} I\left(UDZ^{(i)} \in A\right). \tag{2}$$

This transformation imposes a prioritisation on the integration variables $z_1, \ldots, z_n$. To see this, consider the terms $\sum_{j=1}^{n} U_{ij}d_jz_j$ for $i = 1, \ldots, n$. The L_2 norm of the vector of the coefficients for z_j equals d_j^2. As $d_k^2 \geq d_l^2$ for $k < l$, this means that the L_2 norm of the coefficients for z_k will be greater than or equal to the coefficients of z_l. Also, as Z_j are i.i.d. standard normal random variables, it is very possible that the value for $\sum_{j=1}^{n} U_{ij}d_jZ_j$ will be dominated by the terms $U_{ij}d_jZ_j$ for small values of j.

In the following sections, we will discuss several variance reduction techniques which will be used in the algorithm.

2.1. Conditional Monte Carlo

Conditioning on the last $n-1$ elements of Z in (1), we get

$$p = E\left[P\left(U_{i1}d_1Z_1 + h_i \in A_i, i = 1, \ldots, n \mid Z_{-1}\right)\right],$$

where $Z \sim N_n(0, I)$, $h = (UD)_{-1}Z_{-1}$, $A_i = (a_i, b_i)$. Here, R_{-i} is R with either its ith column or its ith element deleted depending on whether R is a matrix or a vector respectively.

The conditional probability inside the expectation depends only on the variable Z_1 and can be computed explicitly as follows. Each inequality constraint $U_{i1}d_1Z_1 + h_i \in A_i$ requires Z_1 to be in an interval. Thus Z_1 has to be in an intersection of n intervals. Thus there is either no acceptable point for Z_1 or Z_1 will lie between the maximum of the lower bounds and the minimum of the upper bounds of these intervals.

To construct these bounds, we have to take into account the sign of the coefficients of Z_1 in each inequality. The lower bounds are

$$L_i \equiv L_i(Z_{-1}) = \frac{\tilde{a}_i - h_i}{U_{i1}d_1}, \tag{3}$$

and the upper bounds are

$$M_i \equiv M_i(Z_{-1}) = \frac{\tilde{b}_i - h_i}{U_{i1}d_1}, \tag{4}$$

where $\tilde{a}_i$ and $\tilde{b}_i$ are a_i and b_i respectively for a nonnegative value of U_{i1} and their reversed values if otherwise. The conditional probability inside the expectation is

$$\phi(Z_{-1}) = \left[\Phi\left(\min_i M_i\right) - \Phi\left(\max_i L_i\right)\right]^+, \tag{5}$$

where Φ denotes the cumulative distribution function of a standard normal distribution and $x^+ = \max(x, 0)$. The required probability is then written as

$$p = \frac{1}{(2\pi)^{(n-1)/2}} \int \phi(z_{-1})\mathrm{e}^{-\frac{1}{2}\|z_{-1}\|_2^2}\mathrm{d}z_{-1}, \tag{6}$$

where $\|.\|_2$ denotes the L_2 norm. In this expression we have integrated out the Z_1 variable.

Considering this form, we get the conditional MC estimator

$$\hat{p}_{CMC} = \frac{1}{N}\sum_{j=1}^{N}\phi(Z_{-1}^{(j)}), \tag{7}$$

where $Z_{-1}^{(j)}, j = 1, \ldots, N$ are independent standard normal random vectors.

2.2. Importance sampling

A widely used method to reduce the variance of a given estimator is importance sampling (IS). The most common procedure is to choose a parametric family of distributions and minimise the variance of the estimator over this family.

As importance family we use the multivariate normal $N_{n-1}(0, \sigma^2 I_{n-1})$ distributions where $\sigma > 0$ is the parameter of the family. Hence, the elements of Z_{-1} will be independent $N(0, \sigma^2)$ random variables. Given this density, we rewrite expression (6) as

$$p = \frac{1}{(2\pi\sigma^2)^{(n-1)/2}}\int \phi(z_{-1})w\left(z_{-1};\sigma^2\right)\mathrm{e}^{-\frac{1}{2\sigma^2}\|z_{-1}\|_2^2}\mathrm{d}z_{-1},$$

where the importance weights w are given by

$$w\left(z_{-1};\sigma^2\right) = \mathrm{e}^{-\frac{1}{2}\|z_{-1}\|_2^2(1-\frac{1}{\sigma^2})}\sigma^{n-1}. \tag{8}$$

We then sample independent $Z_{-1}^{(1)}, \ldots, Z_{-1}^{(N)} \sim N_{n-1}(0, \sigma^2 I)$ to form the IS estimator

$$\hat{p}_{IS} = \frac{1}{N}\sum_{j=1}^{N}\phi(Z_{-1}^{(j)})w\left(Z_{-1}^{(j)};\sigma^2\right). \tag{9}$$

We would like to use the parameter σ^2 that minimises the variance of the IS estimator.

For unbiased estimators ($\hat{p}_{IS}$ is an unbiased estimator of p), minimising the variance is equivalent to minimising the second moment. The second moment of the IS estimator (9) is given by

$$\frac{1}{(2\pi\sigma^2)^{(n-1)/2}}\int \phi^2(z_{-1})w^2\left(z_{-1};\sigma^2\right)\mathrm{e}^{-\frac{1}{2\sigma^2}\|z_{-1}\|_2^2}\mathrm{d}z_{-1}. \tag{10}$$

As it is analytical intractable to obtain the optimal value of σ^2 using (10), we resort to simulation.

We will use the following iterative procedure. We will first sample independent $Z_1^{(1)}, \ldots, Z_{N_1}^{(1)} \sim N_{n-1}(0, \sigma_0^2 I)$ where $N_1 \in \mathbb{N}$ and $\sigma_0 > 0$. In this paper $\mathbb{N}$ denotes the set of positive integers. We will estimate (10) using importance sampling as

$$\frac{1}{N_1}\sum_{i=1}^{N_1}\phi^2(Z_i^{(1)})w\left(Z_i^{(1)};\sigma_0^2\right)w\left(Z_i^{(1)};\sigma^2\right),$$

and minimise numerically over σ^2 to get an estimator for the optimal value σ_1^2 say. This completes the first iteration. At the kth iteration we will sample independent $Z_1^{(k)}, \ldots, Z_{N_k}^{(k)} \sim N_{n-1}(0, \sigma_{k-1}^2 I)$ and get an estimate for the optimal value of σ^2 by

$$\sigma_k^2 = \arg\min_{\sigma^2}\frac{1}{\sum_{j=1}^{k}N_j}\sum_{j=1}^{k}\sum_{i=1}^{N_j}\phi^2(Z_i^{(j)})w\left(Z_i^{(j)};\sigma_{j-1}^2\right)w\left(Z_i^{(j)};\sigma^2\right).$$

For the last iteration, $k_{\max}$ say, we will have the estimate for the optimal value $\hat{\sigma}^2 = \sigma_{k_{\max}}^2$ and also form the modified IS estimator by

$$\frac{1}{\sum_{j=1}^{k_{\max}}N_j}\sum_{j=1}^{k_{\max}}\sum_{i=1}^{N_j}\phi(Z_i^{(j)})w\left(Z_i^{(j)};\sigma_{j-1}^2\right).$$

2.3. Splitting

As we argued, the first elements of Z are of higher importance than the successive ones. This motivates using splitting to improve the performance of the algorithm.

Splitting is a method that concerns the computation of $E[G(X, Y)]$, where G is a function of the independent random variables X and Y. Denote t_A for the expected time to generate A. We assume that $t_X < t_Y$ and that G is more influenced by X than by Y. In this case, one can reuse the simulated samples from Y and sample from X more often.

This method can be easily fit to our problem if we consider the diagonalisation of the covariance matrix Σ. This decomposition has imposed a prioritisation on the integration variables. In this case we can consider $G(X, Y) \equiv G(Z_{-1}) =$

$\phi(Z_{-1})w(Z_{-1};\hat{\sigma}^2)$, where $X=(Z_2,\ldots,Z_{\gamma+1})'$, $Y=(Z_{\gamma+2},\ldots,Z_n)'$, for some $\gamma<n/2$ and $\hat{\sigma}^2$ is the value obtained in the iterative importance sampling procedure. As computational costs we have $t_X=\gamma$ and $t_Y=n-\gamma-1$.

Given the value of γ, we generate independent $X_{1s},\ldots,X_{Rs}\sim N_\gamma(\mu,\hat{\sigma}^2 I)$ for $s=1,\ldots,S$ and $Y_1,\ldots,Y_R\sim N_{n-\gamma-1}(\mu,\hat{\sigma}^2 I)$ to form the *splitting* estimator given by

$$\hat{p}_{\text{split}}=\frac{1}{RS}\sum_{r=1}^{R}\sum_{s=1}^{S}G(X_{rs},Y_r).$$

Asmussen and Glynn (2007, Chapter V) provide a detailed description of splitting, containing a derivation of an optimal value for S with respect to an efficiency defined as variance per unit time. The optimal efficiency is given by

$$e_\gamma=t_Y\left(\sqrt{t_X/t_Y}+\sqrt{\rho}\right)^2,$$

where $\rho=\text{Corr}\{G(X_{rs_1},Y_r),G(X_{rs_2},Y_r)\}$, $s_1\neq s_2$. The efficiency without any splitting is $n-1$. The optimal value of S is given by $\lfloor\sqrt{t_Y/(\rho t_X)}\rfloor$ where $\lfloor.\rfloor$ denotes the floor function.

We estimate the correlation ρ by $\hat{\rho}$, the Pearson correlation coefficient based on simulations and let

$$S=\begin{cases}\left\lfloor\sqrt{t_Y/(\hat{\rho}t_X)}\right\rfloor & \text{if } e_\gamma\leq n-1\\ 1 & \text{otherwise.}\end{cases}$$

An ad hoc rule is to choose γ as

$$\gamma=\begin{cases}\min\left\{k:\dfrac{\sum_{i=2}^{k}d_i^2}{\sum_{j=2}^{n}d_j^2}>p_v\right\} & \text{if } \dfrac{\sum_{i=2}^{\lfloor\frac{n}{2}\rfloor}d_i^2}{\sum_{j=2}^{n}d_j^2}>p_v\\ \left\lfloor\dfrac{n}{2}\right\rfloor & \text{otherwise,}\end{cases}\tag{11}$$

for some $p_v\in(0.5,1)$. This choice is motivated by the fact that the L_2 norm for the ith column of UD, which corresponds to the coefficients of Z_i, equals d_i^2. An optimal value for γ exists, but it would be computationally intensive to estimate it through simulations. As we will see in Section 4, there is a wide range of splitting points that do almost equally good, so there is no need to hit it precisely.

2.4. Control variates

Control variates is a method that uses a random variable with known expectation which is highly correlated with the quantity that is to be computed.

In this paper we use *multiple controls* (Asmussen and Glynn, 2007, Chapter V) $W=(W_1,\ldots,W_{n_{cv}})$ where $n_{cv}\leq n$ and $W_j=I(\sum_{i=1}^{n}U_{\pi(j)i}d_iZ_i\in A_{\pi(j)})$. Here π is an injective function from $\{1,\ldots,n_{cv}\}$ to $\{1,\ldots,n\}$. These multiple controls can be used within the crude MC estimate (1).

Since applying conditional MC the Z_1 variable is integrated out, we will use a modified version of the control variate W_j. We are going to proceed as in Section 2.1 by conditioning on the last $n-1$ elements of Z, using importance sampling and constructing the control variate as

$$W_j\equiv W_j(Z_{-1})=\left[\Phi\left(b_{\pi(j)}-h_{\pi(j)}\right)-\Phi\left(a_{\pi(j)}-h_{\pi(j)}\right)\right]w\left(Z_{-1};\hat{\sigma}^2\right),$$

where h is as in Section 2.1 and w are the importance weights. The expected value for the control variate W_j, which equals the expected value of the unmodified version, is

$$E_j=\Phi\left(\frac{b_{\pi(j)}}{\sqrt{\sum_{i=1}^{n}U_{\pi(j)i}^2d_i^2}}\right)-\Phi\left(\frac{a_{\pi(j)}}{\sqrt{\sum_{i=1}^{n}U_{\pi(j)i}^2d_i^2}}\right).$$

We will use these types of control variates within the splitting procedure. Given the multiple control vector W, we form a linear regression model with intercept where the responses are the values of $G(Z_r^{(s)})$ and the covariates are $W_i(Z_r^{(s)})-E_i$ for $i=1,\ldots,n_{cv}$. Here $Z_r^{(s)}=(X_{rs}',Y_r')'$. The control variate estimator $\hat{p}_{cv}$ is the least squares estimator of the intercept.

The extra effort of computing the control variate estimator is negligible compared to the total time, and, as we will see in Section 4, in some situations the reduction in variance can be huge.

How to choose the function π? In the next section we give an adaptive method that exploits the random vectors L and M given in (3) and (4) respectively. This adaptive method is prompted by the fact that after conditional Monte Carlo, probability (1) was transformed into form (6) that depends on the random variables $\max L$ and $\min M$. In this way we can see which components of L and M appear more often in (6), and use these components as the range of the function π.

2.5. Pseudocode

In this section we describe the algorithm in pseudocode. All variables are sampled independently.

1. Input parameters: $a, b \in \mathbb{R}^n$ and $\Sigma \in \mathbb{R}^{n\times n}$.
   Tuning parameters: $\mu \in \mathbb{R}$, $\sigma_0 > 0$, $(N_k)_{k=1}^{k_{\max}} \subset \mathbb{N}$, $N \in \mathbb{N}$, $M \in \mathbb{N}$, $R \in \mathbb{N}$, $p_v \in (0.5, 1)$, $n_{cv} \in \{1, \ldots, n\}$.
2. Diagonalise $\Sigma = UD^2U'$ where D^2 is the diagonal matrix of the eigenvalues of Σ in decreasing order and the columns of U are the corresponding orthonormal eigenvectors.
3. **Importance sampling**
   (a) For $k := 1$ to $k_{\max}$
   Generate $Z_1^{(k)}, \ldots, Z_{N_k}^{(k)} \sim N_{n-1}(\mu, \sigma_{k-1}^2 I)$.
   Using numerical minimisation, let
   $$\sigma_k^2 = \arg\min_{\sigma^2} \frac{1}{\sum_{j=1}^{k} N_j} \sum_{j=1}^{k} \sum_{i=1}^{N_j} \phi^2(Z_i^{(j)}) w_j(Z_i^{(j)}) w\left(Z_i^{(j)}; \sigma^2\right),$$
   where $w_j(\cdot) = w(\cdot; \sigma_{j-1}^2)$ is as in (8) and ϕ is as in (5).
   (b) $\hat{\sigma}^2 = \sigma_{k_{\max}}^2$.
   (c) Generate $Z_1, \ldots, Z_N \sim N_{n-1}(\mu, \hat{\sigma}^2 I)$.
   (d) Importance sampling estimator
   $$\hat{p}_{IS} = \frac{1}{N} \sum_{j=1}^{N} \phi(Z_j) w\left(Z_j; \hat{\sigma}^2\right).$$
4. **Splitting**
   (a) Let γ as in (11).
   (b) Let ρ be the Pearson correlation coefficient of the vectors $(G(X_{i1}, Y_i))_{i=1}^M$ and $(G(X_{i2}, Y_i))_{i=1}^M$ where $G(Z) = \phi(Z)w(Z; \hat{\sigma}^2)$, $Z = (X', Y')'$, $X_{1j}, \ldots, X_{Mj} \sim N_\gamma(\mu, \hat{\sigma}^2 I)$ for $j = 1, 2$ and $Y_1, \ldots, Y_M \sim N_{n-\gamma-1}(\mu, \hat{\sigma}^2 I)$.
   (c) Let
   $$S = \begin{cases} \left\lfloor \sqrt{t_Y/(\rho t_X)} \right\rfloor & \text{if } t_Y \left(\sqrt{t_X/t_Y} + \sqrt{\rho}\right)^2 \le t_Y + t_X \\ 1 & \text{otherwise,} \end{cases}$$
   where $t_X = \gamma$ and $t_Y = n - \gamma - 1$.
   (d) Generate $X_{1s}, \ldots, X_{Rs} \sim N_\gamma(\mu, \hat{\sigma}^2 I)$ for $s = 1, \ldots, S$ and $Y_1, \ldots, Y_R \sim N_{n-\gamma-1}(\mu, \hat{\sigma}^2 I)$.
   (e) Splitting estimator
   $$\hat{p}_{\text{split}} = \frac{1}{RS} \sum_{r=1}^{R} \sum_{s=1}^{S} G(X_{rs}, Y_r).$$
5. **Control variates**
   (a) Let $P^{(rs)} = \arg\max_i L_i(Z_r^{(s)})$ and $Q^{(rs)} = \arg\min_i M_i(Z_r^{(s)})$ where L and M are as in (3) and (4) for $r = 1, \ldots, R$, $s = 1, \ldots, S$ and $Z_r^{(s)} = (X_{rs}', Y_r')'$.
   (b) Let $J = \{P^{(rs)}, Q^{(rs)}\}$ for $r = 1, \ldots, R$ and $s = 1, \ldots, S$ and take K (the range of the function π) as the set of the first n_{cv} most frequent values of J.
   (c) Let $W_i(Z_r^{(s)}) = [\Phi(b_i - h_i^{(s)}) - \Phi(a_i - h_i^{(s)})] w(Z_r^{(s)})$ for $i \in K$, where $h_i^{(s)} = \sum_{j=2}^{n} U_{ij} d_j Z_{r(j-1)}^{(s)}$.
   (d) Let $E_i = \Phi(b_i / \sqrt{\sum_{j=1}^{n} U_{ij}^2 d_j^2}) - \Phi(a_i / \sqrt{\sum_{j=1}^{n} U_{ij}^2 d_j^2})$.
   (e) Form a linear regression model with intercept where the responses are the values of $G(Z_r^{(s)})$ and the covariates are $W_i(Z_r^{(s)}) - E_i$ for $i = 1, \ldots, n_{cv}$. The control variate estimator $\hat{p}_{\text{cv}}$ is the least squares estimator of the intercept.

3. Previous methods

There have been many previous approaches for computing multivariate normal probabilities. The most important one is a method proposed by Genz (1992) which is implemented in R and Matlab. Another approach is the so-called Deak's decomposition (Deak, 1980). In this section we will give a brief description of these algorithms and compare them to our method. For a thorough study of the problem, see Genz and Bretz (2009) and the references therein.

3.1. Genz's method

The method of Genz (1992) and Genz and Bretz (2002) starts with a transformation on the integration variables $x = Cz$, where CC' is the Cholesky decomposition of the covariance matrix Σ. The probability is transformed to

$$p = \frac{1}{(2\pi)^{n/2}} \int_{Cz \in A} \mathrm{e}^{-\frac{1}{2}z'z} \mathrm{d}z.$$

The matrix C is lower triangular and the linear constraints $a \leq Cz \leq b$ imply

$$\tilde{a}_i \equiv \frac{1}{c_{ii}} \left(a_i - \sum_{j=1}^{i-1} c_{ij} z_j \right) \leq z_i \leq \frac{1}{c_{ii}} \left(b_i - \sum_{j=1}^{i-1} c_{ij} z_j \right) \equiv \tilde{b}_i, \quad i = 1, \ldots, n.$$

The method proceeds by conditioning on the first $n-1$ elements $z_1, \ldots, z_{n-1}$ and get

$$p = \frac{1}{(2\pi)^{(n-1)/2}} \int \left(\prod_{i=1}^{n-1} I \left\{ z_i \in \left(\tilde{a}_i, \tilde{b}_i \right) \right\} \right) \left[\Phi(\tilde{b}_n) - \Phi(\tilde{a}_n) \right] \mathrm{e}^{-\frac{1}{2}\|z_{-n}\|_2^2} \mathrm{d}z_{-n}.$$

Given this form, the method continues by dynamic importance sampling using the truncated normal distribution as the importance density.

To see this, consider the first linear constraint $z_1 \in (a_1, b_1)$. Then Z_1 is sampled from $TN(a_1, b_1)$ where $TN(\alpha, \beta)$ denotes a standard normal distribution conditioned on the random variable being in the interval (α, β). Given the value for z_1 and considering the second linear constraint $z_2 \in (\tilde{a}_2, \tilde{b}_2)$, one can sample Z_2 from $TN(\tilde{a}_2, \tilde{b}_2)$. The method proceeds dynamically by sampling independently Z_i from $TN(\tilde{a}_i, \tilde{b}_i)$ for $i = 1, \ldots, n-1$.

One could say that this method uses as samples one-dimensional subspaces $\{(Z_1, \ldots, Z_{n-1}, \lambda) : \lambda \in \mathbb{R}\}$ instead of simple points of $\mathbb{R}^n$. It uses a prioritisation on the integration variables and also randomised Quasi MC methods are applied; see Genz and Bretz (2009) for details.

3.2. Deak's method

The method of Deak (1980) also starts with a Cholesky decomposition and uses the transformation $x = Cz$. Then z is expressed as $z = \chi s$ with $\|s\|_2 = 1$ and $\chi \geq 0$, yielding

$$p = \frac{1}{(2\pi)^{n/2}} \int_{\|s\|_2 = 1} \int_{\rho_1(s)}^{\rho_2(s)} \chi^{n-1} \mathrm{e}^{-\frac{\chi^2}{2}} \mathrm{d}\chi \, \mathrm{d}s,$$

where

$$\rho_1(s) = \min \{ r : r \geq 0, a \leq rCs \leq b \}, \qquad \rho_2(s) = \max \{ r : r \geq 0, a \leq rCs \leq b \}.$$

The innermost one-dimensional integral can be handled by numerical integration based on the chi-square distribution and a Monte Carlo estimator is constructed by sampling independently points from the unit sphere $\|s\|_2 = 1$.

This is equivalent to say that samples are elements of the one-dimensional subspaces $\{\lambda(Z_1, \ldots, Z_n) : \lambda \in \mathbb{R}\}$.

3.3. Our method

In our estimator, the samples are the one-dimensional subspaces $\{(\lambda, Z_2, \ldots, Z_n) : \lambda \in \mathbb{R}\}$. Importance sampling is used for which samples are taken from the $N_{n-1}(0, \sigma^2 I)$ distribution. The parameter σ^2 is calibrated adaptively in the algorithm. The diagonalisation of the covariance matrix Σ forces a prioritisation on the integration variables which allows us to use the splitting procedure. Finally, control variates are also used.

4. Simulation results

In this section, we report the results using four different covariance matrices and several rectangular regions. We will compare our proposed method to the algorithm of Genz implemented in the MATLAB function qsimvnv.

We consider covariance matrices of the form $\Sigma = BB'$, where B is one of the following:

- $L \in \mathbb{R}^{54 \times 54}$ taken from the first example of Lin et al. (2002),
- $N \in \mathbb{R}^{200 \times 200}$ where N_{ij} are independent realisations of $N(0, 4)$,
- $E \in \mathbb{R}^{1000 \times 1000}$ where E_{ij} are independent realisations of $N(\mathrm{Exp}(1), 1)$,
- $K = (1, I) \in \mathbb{R}^{1000 \times 1001}$ where $1 = (1, \ldots, 1)'$ and I is the identity matrix. This matrix is used in multiple comparisons (Dunnett, 1955).

The first and last matrices are deterministic, while the other two are fixed realisations drawn from the above distributions. The distribution of their singular values is plotted in Fig. 1.

We consider rectangular regions of the form $(a, b) = (-c, c)$, i.e. $-a_i = b_i = c$, for some threshold $c > 0$. This is the kind of probabilities that are needed for the computation of the p-value of the tests described in Lin et al. (2002) and also

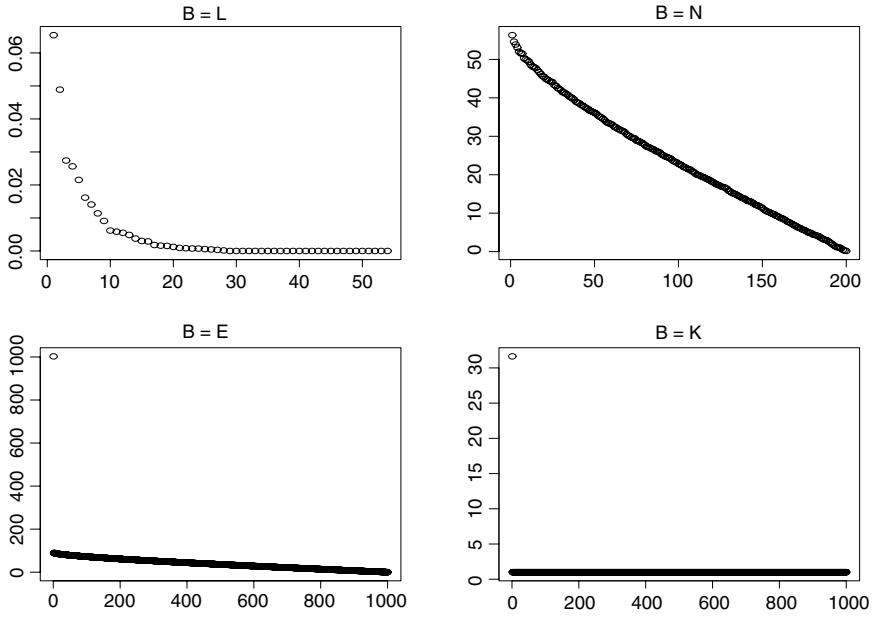

Fig. 1. Singular values of the matrices B we use in our simulations.

for Dunnett (1955). We report in the tables estimates for

$$q = P\left(\|BZ\|_\infty > c\right) = 1 - P\left(-c, c, BB'\right),$$

where $\|x\|_\infty = \max_i |x_i|$ for $x \in \mathbb{R}^n$. By replacing ϕ by $1-\phi$ in Section 2, we get estimators $\hat{q}_{MC}$, $\hat{q}_{CMC}$, $\hat{q}_{IS}$, $\hat{q}_{split}$ and $\hat{q}_{cv}$ for q. Genz's method estimates p and we report one minus this estimate.

The running times of all methods are roughly comparable. For the simulations, for each threshold c, we have used 10^3 runs. Each run is based on 10^4 samples for the IS estimator while for splitting we have used a sample to equalise the computational effort with IS. For calibration of the importance sampling parameter we have used 3000 samples ($N_1 = 300$, $N_j = \min(2N_{j-1}, 3000 - \sum_{k=1}^{j-1} N_k)$ for $j \in \{2, 3, 4\}$). We have set $p_v = 0.85$. The control variates are used within the splitting estimator with $n_{cv} = 10$. We ran our method in R. Genz's method based on a standard implementation in the MATLAB function qsimvnv with 10^4 samples. This method uses randomised quasi-random numbers and is unbiased. We did not use the implementation of Genz's method in the R package mvtnorm (Genz and Bretz, 2009; Genz et al., 2010) because it showed bias in high dimensions.

Before we move to simulation results, we will evaluate our ad hoc choice for the splitting point γ. We consider the plots in Fig. 2. These plots contain the relative efficiency of all points for the four matrices and certain thresholds. One can see that our ad hoc rule is not far away from the minimum point and is also better than with no splitting.

All procedures except $\hat{q}_{cv}$ are, theoretically, unbiased. We checked that the bias for the $\hat{q}_{cv}$ is also negligible. Therefore we use the standard deviation for comparisons. In Tables 1–4 we report the standard deviation based on 10^3 runs. We will also report the true value q based on the crude Monte Carlo estimator (2) using 10^8 samples.

Starting with the real data matrix from Lin et al. (2002), Table 1, we see the significance of the four variance reduction techniques applied to our algorithm. For the smaller threshold, we were slightly better than Genz's method when control variates were included, but for the larger ones we were significantly superior even with the conditional MC estimate. There was an improvement of the standard deviation by a factor of around two with the splitting procedure compared to IS for all thresholds. The improvement brought by control variates is particularly dramatic for the largest threshold $c = 0.09$; a reduction by a factor of 10 of the standard deviation compared with the splitting estimator and a factor of around 300 compared with Genz's estimator. Also a reduction by a factor of 10 and a factor of 6 was achieved comparing the IS and the conditional MC respectively with Genz's estimator. For the two smaller thresholds there was no improvement from IS compared to conditional MC. This is because, after conditional MC, we may no longer deal with rare events. The crude MC estimator was outperformed by every other estimator.

The N and E matrices, Tables 2 and 3, show the significance of the distribution of the singular values. Our procedure performed much better with the E matrix (the first singular value is much greater than the rest) than with the N matrix. This is easily seen when one compares the results of the crude and the conditional MC estimator. Genz's method was better with N for all three thresholds. However, for E, our method performed better for all estimators and especially for the higher thresholds where a considerable reduction was achieved with conditional MC and splitting.

For the K matrix, Table 4, our method performed better than Genz's method for all thresholds. There was a reduction of standard deviation by a factor of around 8 and 20 for $c = 7$ and $c = 8.5$ respectively. This matrix again showed the

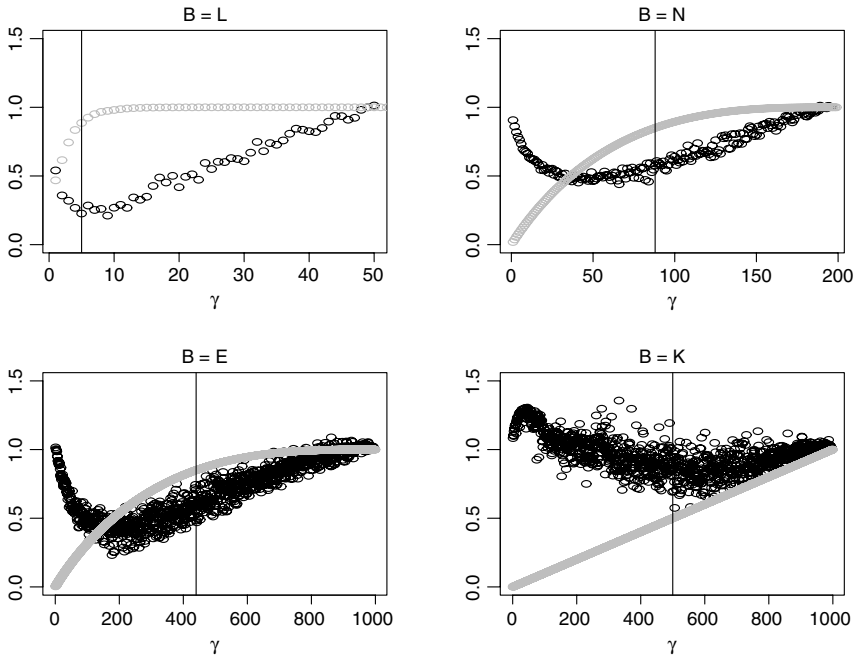

Fig. 2. The black points denote the relative efficiency $\frac{e_\gamma}{n-1}$ (=1 is the value with no splitting) based on simulations, the grey points denote the ratio $\frac{\sum_{j=2}^{\gamma} d_j^2}{\sum_{j=2}^{n} d_j^2}$ and the vertical line denotes the splitting point γ computed via the ad hoc rule (11).

Table 1
Standard deviation of the estimates for $q = P(\|LZ\|_\infty > c)$.

c	0.038	0.07	0.09
q	0.1013	2.02×10^{-4}	0.101×10^{-5}
$\hat{q}_{cv}$	0.43×10^{-3}	0.03×10^{-4}	0.0009×10^{-5}
$\hat{q}_{split}$	0.71×10^{-3}	0.07×10^{-4}	0.0101×10^{-5}
$\hat{q}_{IS}$	1.28×10^{-3}	0.17×10^{-4}	0.0195×10^{-5}
$\hat{q}_{CMC}$	1.28×10^{-3}	0.17×10^{-4}	0.0335×10^{-5}
$\hat{q}_{MC}$	3.07×10^{-3}	1.40×10^{-4}	1.1325×10^{-5}
Genz	0.51×10^{-3}	0.46×10^{-4}	0.2632×10^{-5}

Table 2
Standard deviation of the estimates for $q = P(\|NZ\|_\infty > c)$.

c	80	100	130
q	0.5820	0.0826	11.7×10^{-4}
$\hat{q}_{cv}$	3.02×10^{-3}	1.66×10^{-3}	1.74×10^{-4}
$\hat{q}_{split}$	3.12×10^{-3}	1.72×10^{-3}	1.77×10^{-4}
$\hat{q}_{IS}$	4.14×10^{-3}	2.19×10^{-3}	2.48×10^{-4}
$\hat{q}_{CMC}$	4.30×10^{-3}	2.40×10^{-3}	2.74×10^{-4}
$\hat{q}_{MC}$	4.86×10^{-3}	2.79×10^{-3}	3.41×10^{-4}
Genz	1.06×10^{-3}	0.69×10^{-3}	0.62×10^{-4}

significance of the conditional MC with a high first singular value, but as expected from Figs. 1 and 2, there was a negligible improvement through splitting and no improvement from control variates.

5. Concluding remarks

We have compared different methods for computing multivariate normal probabilities: our method implemented in R and Genz's procedure implemented in MATLAB. The rectangular regions used were of the form $(a, b) = (-c, c)$, i.e. $-a_i = b_i = c$ for some threshold $c > 0$.

Simulation results have shown that our method can perform better than Genz's method in some situations. For some specific cases, as e.g. in Table 1, we can dramatically outperform this method.

In all cases we have seen the contribution of the four variance reduction techniques. For specific covariance matrices there is a significant improvement using conditional MC. On the other hand, the gain from IS can be negligible since due to

Table 3
Standard deviation of the estimates for $q = P(\|EZ\|_\infty > c)$.

c	150	250	300
q	0.8244	48.5×10^{-4}	5.95×10^{-5}
$\hat{q}_{cv}$	1.32×10^{-3}	1.00×10^{-4}	0.56×10^{-5}
$\hat{q}_{split}$	1.32×10^{-3}	1.03×10^{-4}	0.59×10^{-5}
$\hat{q}_{IS}$	1.67×10^{-3}	1.47×10^{-4}	0.87×10^{-5}
$\hat{q}_{CMC}$	1.81×10^{-3}	1.47×10^{-4}	0.90×10^{-5}
$\hat{q}_{MC}$	2.73×10^{-3}	6.99×10^{-4}	5.73×10^{-5}
Genz	1.37×10^{-3}	2.46×10^{-4}	1.96×10^{-5}

Table 4
Standard deviation of the estimates for $q = P(\|KZ\|_\infty > c)$.

c	6	7	8.5
q	0.0101	51.4×10^{-5}	16.9×10^{-7}
$\hat{q}_{cv}$	1.04×10^{-4}	1.23×10^{-5}	2.01×10^{-7}
$\hat{q}_{split}$	1.09×10^{-4}	1.24×10^{-5}	2.05×10^{-7}
$\hat{q}_{IS}$	1.24×10^{-4}	1.44×10^{-5}	2.32×10^{-7}
$\hat{q}_{CMC}$	1.29×10^{-4}	1.48×10^{-5}	2.35×10^{-7}
$\hat{q}_{MC}$	10.3×10^{-4}	22.7×10^{-5}	124×10^{-7}
Genz	3.55×10^{-4}	9.11×10^{-5}	42.8×10^{-7}

the use of conditional MC we may no longer deal with rare events. In most examples splitting procedure led to improvements and also, in some situations, control variates can lead to dramatic improvements.

Appendix. Supplementary data

Supplementary material related to this article can be found online at doi:10.1016/j.csda.2010.10.005.

References

Anderson, J., Pemberton, J., 1985. The grouped continuous model for multivariate ordered categorical variables and covariate adjustment. Biometrics 41 (4), 875–885.
Asmussen, S., Glynn, P., 2007. Stochastic Simulation: Algorithms and Analysis. Springer.
Deak, I., 1980. Three digit accurate multiple normal probabilities. Numerische Mathematik 35 (4), 369–380.
Dunnett, C., 1955. A multiple comparison procedure for comparing several treatments with a control. Journal of the American Statistical Association 50 (272), 1096–1121.
Genz, A., 1992. Numerical computation of multivariate normal probabilities. Journal of Computational and Graphical Statistics 1, 141–149.
Genz, A., 1993. Comparison of methods for the computation of multivariate normal probabilities. Computing Science and Statistics 25, 400–405.
Genz, A., Bretz, F., 2002. Comparison of methods for the computation of multivariate t probabilities. Journal of Computational and Graphical Statistics 11 (4), 950–971.
Genz, A., Bretz, F., 2009. Computation of Multivariate Normal and t Probabilities. Springer Verlag.
Genz, A., Bretz, F., Miwa, T., Mi, X., Leisch, F., Scheipl, F., Hothorn, T., 2010. mvtnorm: Multivariate normal and t distributions. R package version 0.9-9.
Lin, D., Wei, L., Ying, Z., 2002. Model-checking techniques based on cumulative residuals. Biometrics 58 (1), 1–12.
Miwa, T., Hayter, A., Kuriki, S., 2003. The evaluation of general non-centred orthant probabilities. Journal of the Royal Statistical Society. Series B. Statistical Methodology 65 (1), 223–234.
Ochi, Y., Prentice, R., 1984. Likelihood inference in a correlated probit regression model. Biometrika 71 (3), 531–543.
R Development Core Team. 2010. R: A Language and Environment for Statistical Computing. R Foundation for Statistical Computing, Vienna, Austria. ISBN 3-900051-07-0.
Sandor, Z., Andras, P., 2004. Alternative sampling methods for estimating multivariate normal probabilities. Journal of Econometrics 120 (2), 207–234.
Schervish, M., 1984. Multivariate normal probabilities with error bound. Applied Statistics 33, 81–87.
Somerville, P., 1998. Numerical computation of multivariate normal and multivariate-t probabilities over convex regions. Journal of Computational and Graphical Statistics 7 (4), 529–544.
Szantai, T., 2000. Improved bounds and simulation procedures on the value of the multivariate normal probability distribution function. Annals of Operations Research 100 (1), 85–101.
Vijverberg, W., 1997. Monte Carlo evaluation of multivariate normal probabilities. Journal of Econometrics 76 (1), 281–307.
Wang, M., Kennedy, W., 1992. A numerical method for accurately approximating multivariate normal probabilities. Computational Statistics and Data Analysis 13 (2), 197–210.