Dimension reduction in the computation of value-at-risk

Claudio Albanese[1] Ken Jackson[2] Petter Wiberg[3]

February 28, 2002

[1]Department of Mathematics, University of Toronto, 100 St. George Street, Toronto, Ontario, Canada M5S 3G3 , and Math Point Ltd., Toronto (www.math-point.com). e-mail: albanese@math.toronto.edu, phone: 416-978-5237

[2]Department of Computer Science, University of Toronto, 10 King's College Road, Toronto, Ontario, Canada M5S 3G4. e-mail: krj@cs.toronto.edu, phone: 416-978-7075

[3]Communicating author. Department of Computer Science, University of Toronto, 10 King's College Road, Toronto, Ontario, Canada M5S 3G4. e-mail: wiberg@cs.toronto.edu, phone: 416-978-3905

Abstract

Value-at-risk models can have many dimensions. We present two new algorithms for dimension reduction in value-at-risk algorithms with $\Delta\Gamma$ approximations. In the first method, we compute a reduced portfolio with a small mean square error for the residual and, in the second method, we use low rank approximations to find a reduced portfolio. The paper concludes with an example, estimating value-at-risk and hedging an option portfolio, with that demonstrates that dimension reduction leads to large savings in computational time without sacrificing accuracy.

1 Introduction

In this paper we present two new portfolio dependent methods for dimension reduction in models for market risk, the risk of a decrease in the value of a portfolio due to adverse market movements. Banks are required by regulators to use mathematical models for *value-at-risk* to estimate their exposure to market risk [3, 15, 18]. To cover potential losses, capital charges are determined based on the result from these simulations and on the quality of the banks' models [2, 3, 4].

Value-at-risk (VaR) is the maximum loss over a given holding period and with a specified confidence level α, or equivalently it is the $(1-\alpha)$-quantile of the distribution for the gain. For linear portfolios and normally distributed market factors, the problem is well understood and there is a closed form solution for the VaR (see [18, 15]). The general problem poses two main challenges. First, there is ample evidence that actual return distributions have fat tails, and normal distributions therefore lead to serious underestimation of the risk of large losses. Constructing realistic models for return and calibrating these models from historical returns is an important problem, and it has significant impact on the conclusions drawn about the market risk for a portfolio [5, 8]. Second, the value of derivative securities, a security for which the value is contingent on the value of some market factor, for example options, are non-linear functions of the market factors. To accurately model the risk for derivative portfolios the convexity must be captured by the model. The non-linearity combined with a high number of dimensions, makes it an interesting problem for which a number of methods have been suggested, see for example [1, 6, 8, 9, 10, 13, 15, 17, 18, 22].

The value of a portfolio is a non-linear function of a set of risk factors. For short holding periods, such as a few days, the gain for a portfolio can be approximated by the first two terms of the Taylor series:

$$\text{gain for portfolio} \approx \Pi = \frac{1}{2} y^T \Gamma y + y^T \Delta + \Xi, \tag{1}$$

where Γ is the symmetric $n \times n$ matrix of second order partial derivatives, Δ is the n vector of first order partial derivatives and Ξ is the gain for $y = 0$. The n-vector of random variables y is the returns on the risk factors. In the finance literature, equation (1) is traditionally called a $\Delta\Gamma$ *approximation*. If the time horizon is relatively short, a number of days as opposed to months, the distribution of Π can be used to find an approximation to the VaR [1]:

$$\mathrm{prob}\,\{\Pi - \mathrm{E}[\Pi] \leq \mathrm{VaR}\,\} = 1 - \alpha. \tag{2}$$

The random variables y are distributed according to some multivariate stochastic model with parameters estimated from time series with market data.

For many portfolios, a few main directions determine the main behavior of the gain, because of the combined effect of correlation of the market factors and the quantities of each security held in the portfolio. Therefore, it is natural to search for a simpler approximation depending on fewer factors $k \ll n$ that is close to the original model. This problem has been examined by other authors, e.g., Hull [15] shows how to use principal component analysis in an interest rate model; Kreinin et al. [16] present a principal component based method for linear portfolios with normally distributed risk factors; and Reimers and Zerbs [19] develop a reduction method where asset blocks are represent by their dominant principal components and cross block covariances by the covariance for the largest principal component of each block. Our method uses higher order information about the convexity of the gain, and we also discuss how to handle non-normal returns.

To be more specific, the goal is to find a projection[1] $\mathcal{P}$ onto a subspace spanned by the orthogonal columns of the $n \times k$ matrix Q_1. The idea is to choose $\mathcal{P}$ (or Q_1) such that

$$\Pi(y) \approx \Pi(\mathcal{P}y) = \Pi_1(z_1).$$

The vectors of random variables y and z_1 satisfy $\mathcal{P}y = Q_1 z_1$. We present two methods for this problem. The first method solves the problem by finding a lower dimension approximation with a small mean square error. The second method uses the identification of quadratic forms with matrices and solves the problem, after re-scaling the variables, by finding a lower rank matrix close to the original $\Delta\Gamma$ approximation.

In Section 2, we review the fast convolution method for value-at-risk with particular focus on the estimation and algebraic transformations of the portfolio, a topic that was hinted at briefly in [1]. Section 3 is the main section of the paper and presents the two methods for dimension reduction. In Section 4, we follow up on the theoretical arguments of Section 3 with a computational example which illustrates how the dimension reduction influences the estimate for value-at-risk, and we show that, for the option portfolio considered, almost all risk is captured by a few dimensions. We also discuss hedging with the reduced model, and show that the accuracy is good. To find the hedge we must solve an optimization problem, and the example shows that by using a reduced model the computation time is only a fraction of the time to solve the full problem. In the final section, we discuss conclusions and future work.

2 The fast convolution algorithm for value-at-risk

The purpose of this section is two fold. We review the steps of the fast convolution method for value-at-risk [1] that is used for the numerical example in the Section 4. Furthermore, in contrast to [1], we give an in depth presentation of the portfolio dependent estimation and the transformations of the $\Delta\Gamma$ approximation used in the algorithm.

The input to the algorithm is a time series matrix X and the gain function for a portfolio. The columns of X are historical time series of daily prices for each of the risk factors in the model. The gain function is represented as a $\Delta\Gamma$ approximation (1). The

[1] A *projection* is a Hermitian matrix $\mathcal{P}$ such that $\mathcal{P}^2 = \mathcal{P}$.

output from the algorithm is the value-at-risk, i.e., the solution to the equation (2). An important feature of the fast convolution method is that the gradients of value-at-risk can be computed efficiently. We refer to [1] for details.

2.1 Estimating the mean and covariance for returns

For a time series matrix X stretching over $d+1$ dates, the element x_{ij} is the price of the jth risk factor on the date i. To avoid complications for prices with different scales and for risk factors measured in different units, for example stock prices and interest rates, it is preferable to model the returns on the risk factors, rather than to model the prices of the risk factors directly. Therefore, we introduce a $d \times n$ matrix Y, with elements defined by

$$y_{ij} = \ln x_{ij} - \ln x_{i-1j},$$

containing the time series of continuously compounded returns.

The sample means of the returns are

$$m_j = \frac{1}{d} \sum_{i=1}^{d} y_{ij},$$

and sample means for the matrix Z defined by shifted returns $z_{ij} = y_{ij} - m_j$ are zero. The mean of daily returns is typically close to zero.

It has been observed that the statistics of financial time series tend to change over time. For estimation, this means that old data may not carry much relevant information and more recent data should be given more importance. A variety of estimators for the covariance matrix C have been proposed in the finance literature (see for example [7, 15, 21]). From an algorithmic point of view, many of the estimators are of the form

$$C = Z^T W Z, \tag{3}$$

where W is a weight matrix. Two examples are the standard sample covariance estimate, where W is diagonal with elements $1/d$; and the exponentially weighted estimate where W is diagonal with

$$w_{ii} = \left(\frac{1-\lambda}{1-\lambda^d} \right) \lambda^{i-1}.$$

To explicitly form the covariance matrix C may be informative, but in algorithms it is often preferable, for both reasons of efficiency and of numerical accuracy, to use the above form for the equation of the estimator, if it is available. For example, if $d < n$, the C matrix is singular and this problem can be avoided. Furthermore, if the time series are highly correlated, forming C may lead to loss of accuracy and C is more ill-conditioned than Z.

2.2 Portfolio decomposition and portfolio dependent estimation

In the special case that the risk factor model for returns y is a multivariate normal distribution, it is uniquely determined by the mean m and the covariance matrix C. However, in general, multivariate distributions are not uniquely determined by their marginal distributions, and it is easy to construct examples that show that the dependence structure may have a large impact on the risk measure [11, 12]. To accurately model the distribution in the directions that are important for the portfolio, we propose a portfolio dependent procedure for estimating the remaining parameters.

First, the $\Delta\Gamma$ approximation (1) can be translated to make it a function of the new random variables $z = y - m$ which have mean zero:

$$\begin{aligned}\Pi &= \frac{1}{2} z^T \Gamma z + z^T(\Delta + \Gamma m) + (\Xi + m^T\Delta + \frac{1}{2} m^T \Gamma m) \\ &= \frac{1}{2} z^T \Gamma z + z^T \widetilde{\Delta} + \widetilde{\Xi}.\end{aligned} \tag{4}$$

The impact of time, on the value of a derivatives portfolio, even for a short holding period, may be substantial. The first order effect of time can be included in the $\Delta\Gamma$ approximation by taking Ξ in (1) to be the "Theta" of the portfolio, the partial derivative of the value with respect to time,

$$\Xi = T\Theta = T\frac{\partial V}{\partial t}(S_0, 0),$$

where T is the holding period. But, since

$$\mathrm{E}[\Pi] = \frac{1}{2}\left[\operatorname{trace}(C\Gamma) + m^T\Gamma m\right] + m^T\Delta + \Xi,$$

Ξ vanishes in the equation (2) used to approximate value-at-risk. Since Ξ cancels, we shall assume that Ξ is zero unless otherwise stated. Similarly, the deterministic gain from the passage of time can be neglected since any higher order time derivative terms included in the approximation cancel (mixed derivative terms do of course not cancel since they are not deterministic).

The $\Delta\Gamma$ approximation (4) can be decomposed as a sum of portfolio marginals:

$$\Pi = \sum_{i=1}^{n} \left(\frac{\lambda_i}{2} \overline{z}_i^2 + \overline{\Delta}_i \overline{z}_i \right)$$

where the random variables $\overline{z}$ are *uncorrelated*. The random variables are related by a linear transformation $z = U\overline{z}$, and $\overline{\Delta} = U^T\widetilde{\Delta}$. The matrix U is computed by solving the generalized eigenvalue problem

$$\begin{aligned}\Lambda &= U^T \Gamma U, \\ C &= UU^T,\end{aligned}$$

where Λ is a diagonal matrix with diagonal elements λ_i.

There are several alternative methods for solving this problem. With C formed explicitly, the standard method is to find the Cholesky factorization $C = VV^T$, compute the Schur form

$$Q\Lambda Q^T = V^T \Gamma V$$

and take $U = VQ$. For estimators of the form (3), forming and factorizing C is unnecessary. If $d < n$, we can use the implicit factorization $C = (W^{1/2}Z)^T(W^{1/2}Z)$, and compute the Schur form

$$Q\Lambda Q^T = (W^{1/2}Z)\Gamma(W^{1/2}Z)^T.$$

If on the other hand $d > n$, we can use the QR factorization $VR = W^{1/2}Z$, and compute the Schur form

$$Q\Lambda Q^T = R\Gamma R^T.$$

In the coordinate system U, the Γ matrix is diagonal and the vector of random variables has the identity as the covariance matrix. For a non-normal model, we estimate the remaining parameters for each of the transformed variables independently. This means that we restrict ourselves to models where the components are independent, i.e, the density can be factored as a product of marginal densities,

$$p(\overline{z}) = p_1(\overline{z_1}) \cdots p_n(\overline{z}_n).$$

In the third and final step, the remaining parameters for each marginal density p_i is calibrated using either a maximum likelihood estimator or by fitting the quantiles. In the language of copulas, we choose a copula that lets us fit each marginal in the directions that are the most important for the portfolio.

2.3 Convolutions and computing value-at-risk

After the estimation phase, we have a distribution p_i for each of the independent risk factors $\overline{z}_i$. The impact of $\overline{z}_i$ on the portfolio value is captured by a one dimensional quadratic function,

$$\pi_i = \frac{\lambda_i}{2}\overline{z}_i^2 + \overline{\Delta}_i \overline{z}_i.$$

The gain is the sum of the quadratics

$$\Pi = \sum_{i=1}^{n} \pi_i$$

To compute value-at-risk, we must find the distribution for Π. Let p_i be the density of the quadratic π_i. Since the quadratics π_i are independent, it follows that density for Π is the convolution of the marginal densities:

$$p_\Pi = p_1 * \cdots * p_n.$$

The portfolio density p_Π can be approximated by discretizing the marginal densities p_i, and then multiplying and inverting the product of the FFTs $\widehat{p_i}$. For the discretization, the densities are restricted to a finite interval covered by a regular grid. Since density p_i has a singularity at the critical point of π_i, special care must be taken to ensure that the probability density is preserved. We take

$$p_j^{(i)} = \mathrm{prob}_i \left\{ \pi_i \in [\zeta_j - \Delta\zeta/2, \zeta_j + \Delta\zeta/2] \right\},$$

for each grid point ζ_j. Then, the probability density is approximated by the FFT:

$$p_j^\Pi = \left(\prod_{i=1}^{n} \widehat{p_j^{(i)}} \right)^\vee$$

The computation of derivatives with respect to Δ and Γ is very similar in spirit, and we refer to [1] for details.

The convolution step of the method provides an approximation p_j^Π to the density function for Π at each grid point, and similar approximations for the derivatives in the directions of the coordinate components. To compute value-at-risk, we need to solve the equation

$$\int_{-\infty}^{\mathrm{VaR}\,+\mathrm{E}[\Pi]} p_\Pi(x)dx = 1 - \alpha \tag{5}$$

where α is the confidence level. Since the integrand is known only at the grid points, we use a quadrature rule to approximate the integral. The method to compute the gradients is similar. By differentiating (5) we get an integral that can be approximated with the trapezoidal rule.

The number of floating point operations required by this method is $O(n^3 + nm\log m)$ where n is the number of risk factors and m is the grid size. This does not include the floating point operations in the estimation step. If the standard sample estimators are used for the mean and covariance matrix with d dates, this adds $O(n^2 d)$ floating point operations to the total. Our experiments (see [1]) show that, for a fixed grid size m, the $O(nm\log m)$-term tends to dominate the computation time.

3 Dimension reduction

In this section, we present two methods for dimension reduction for a value-at-risk model. Given a portfolio Π with many risk factors, the objective in dimension reduction is to find a $\Delta\Gamma$ approximation Π_1 that captures the main properties of Π but that with fewer dimensions. Successful dimension reduction greatly reduces the time to compute value-at-risk. This is particularly important when the problem needs to be solved many times, for example in an optimization algorithm.

The change in value for the full portfolio, represented as a $\Delta\Gamma$ approximation, is a function of the form

$$\Pi = \frac{1}{2} y^T \Lambda y + y^T \Delta. \tag{6}$$

The constant term has been left out since it will cancel in (2) anyway. We look for reductions by restricting the dependence in Π_1 to a subspace of the original space of risk factors. Let $\mathcal{P}$ be a projection onto the k dimensional subspace spanned by the orthonormal columns of the $n \times k$ matrix Q_1. Moreover, $\mathcal{P}^\perp$ is the projection onto the complementary subspace spanned by the orthonormal columns of Q_2. Let $z = (z_1, z_2)$ satisfy

$$y = \mathcal{P}y + \mathcal{P}^\perp y = Q_1 z_1 + Q_2 z_2. \tag{7}$$

Based on this factorization of the risk factor space, we define the reduce approximation

$$\Pi(y) \approx \Pi(\mathcal{P}y) = \Pi_1(z_1). \tag{8}$$

In the sections below, we present two methods for finding a projection so that Π and Π_1 are close. Method 1 views the functions as random variables and requires the two functions to be close in a probabilistic sense. Method 2 identifies the functions with matrices and, if the random variables are scaled properly, closeness can be measured with a matrix norm.

3.1 Method 1

To find the approximation Π_1 in (8), we insist that the mean square error $\mathrm{E}[(\Pi - \Pi_1)^2]$ is small. To motivate the strategy for dimension reduction, we need the following lemma:

Lemma 1. *Let A be an $n \times n$ matrix. Suppose that*

$$\max_{\|a\|_2=1} \max_{\|b\|_2=1} \mathrm{E}[(a^T x)^2 (b^T x)^2] \leq \alpha.$$

Then,

$$\mathrm{E}[(x^T A x)^2] \leq n\alpha \|A\|_F^2.$$

Proof. From the Cauchy-Schwartz inequality, it follows that

$$\mathrm{E}[(x^T A x)^2] \leq \mathrm{E}[(x^T x)(x^T A^T A x)].$$

Since $A^T A$ is symmetric, there is an orthogonal matrix Q such that $A^T A = Q^T \Sigma^2 Q$ where Σ is diagonal. Hence,

$$\mathrm{E}[(x^T x)(x^T A^T A x)] = \mathrm{E}[(x^T Q^T Q x)(x^T Q^T \Sigma^2 Q x)].$$

Define $y_i = q_i^T x$ where q_i^T are the rows of Q. By assumption, we have $\mathrm{E}[y_i^2 y_j^2] \le \alpha$ for all i and j, and it follows that

$$\begin{aligned}
\mathrm{E}[(y^T y)(y^T \Sigma^2 y)] &= \mathrm{E}\left[(\sum_k y_k^2)(\sum_l \sigma_l^2 y_l^2)\right] \\
&= \sum_l \sigma_l^2 \sum_k \mathrm{E}[y_k^2 y_l^2] \\
&\le n\alpha \sum \sigma_j^2 .
\end{aligned}$$

Since $\sum \sigma_j^2 = \|A\|_F^2$ the lemma follows. □

Assume that the 4th order moments in the lemma are bounded, and that we have an estimate for this bound α. Consider a tolerance $\epsilon > 0$. By re-ordering the components of y, we can order the eigenvalues, the diagonal elements of Λ, such that $|\lambda_i| \ge |\lambda_{i+1}|$ for all i. Let k be the smallest index for which

$$\sum_{i=k+1}^{n} \lambda_k^2 \le \epsilon \tag{9}$$

Partition $y = (y_1, y_2)$ where y_1 contains the first k coordinates of y. We can then write (6) as

$$\begin{aligned}
\Pi =& \frac{1}{2}\left[y_1^T, 0\right] \begin{bmatrix} \Lambda_1 & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} y_1 \\ 0 \end{bmatrix} + \left[y_1^T, 0\right] \begin{bmatrix} \Delta_1 \\ 0 \end{bmatrix} \\
&+ \frac{1}{2}\left[0, y_2^T\right] \begin{bmatrix} 0 & 0 \\ 0 & \Lambda_2 \end{bmatrix} \begin{bmatrix} 0 \\ y_2 \end{bmatrix} + \left[0, y_2^T\right] \begin{bmatrix} 0 \\ \Delta_2 \end{bmatrix} .
\end{aligned}$$

The contribution to the mean square error from Λ_2 is small relative to ϵ, but the effect on the gain from Δ_2 may still be large. With a simple trick we can keep all the information in Δ_2. Let $V = [v_1, V_2]$ be an orthogonal matrix with the first column $v_1 = \Delta_2 / \|\Delta_2\|_2$. Since Δ_2 is orthogonal to the column vectors in V_2, we define the $(k+1)$-vector z_1 and the $(n-k-1)$-vector z_2 in (7) by

$$z_1 = \begin{bmatrix} y_1 \\ v_1^T y_2 \end{bmatrix} \quad \text{and} \quad z_2 = V_2^T y_2 .$$

Hence, the reduced $\Delta\Gamma$ approximation is

$$\Pi_1(z_1) = \frac{1}{2} z_1^T \begin{bmatrix} \Lambda_1 & 0 \\ 0 & v_1^T \Lambda_2 v_1 \end{bmatrix} z_1 + z_1^T \begin{bmatrix} \Delta_1 \\ \|\Delta_2\|_2 \end{bmatrix} .$$

This dimension reduction method is summarized in Algorithm 1.

To relate the reduced $\Delta\Gamma$ approximation back to the mean square error, we observe that the residual has a quadratic term only:

$$\Pi - \Pi_1 = \frac{1}{2}[z_1^T, z_2^T] \left[\begin{array}{cc|c} 0 & 0 & 0 \\ 0 & 0 & v_1^T \Lambda_2 V_2 \\ \hline 0 & V_2^T \Lambda_2 v_1 & V_2^T \Lambda_2 V_2 \end{array}\right] \begin{bmatrix} z_1 \\ z_2 \end{bmatrix} .$$

Input: A mean n-vector m, an $n \times n$ covariance matrix C, an n-vector $\mathbf{\Delta}$, an $n \times n$ matrix Γ, a tolerance $\epsilon > 0$ and a bound for the moments α.

Output: A new $\Delta\Gamma$ approximation Π_1 with fewer dimensions, and the vector v_1 that characterizes the transformation of the risk factors.

- Translate the $\Delta\Gamma$ approximation relative to the mean,

$$\Delta \leftarrow \Delta + \Gamma m.$$

- Solve the eigenvalue problem

$$\Lambda = U^T \Gamma U,$$
$$C = UU^T.$$

  Order the eigenvalues so that $|\lambda_i| \geq |\lambda_{i+1}|$, and let $\Delta \leftarrow U^T \Delta$.

- Find the smallest integer k such that

$$\sum_{i=k+1}^{n} \lambda_k^2 \leq \epsilon.$$

- Compute Π_1:

$$\begin{aligned}
\Delta_1 &\leftarrow [\Delta(1\!:k), \|\Delta(k+1\!:n)\|_2], \\
v_1 &\leftarrow \Delta(k+1\!:n)/\|\Delta(k+1\!:n)\|_2, \\
\Lambda_1 &\leftarrow \begin{bmatrix} \Lambda(1\!:k, 1\!:k) & 0 \\ 0 & v_1^T \Lambda(k+1\!:n, k+1\!:n) v_1 \end{bmatrix}.
\end{aligned}$$

Algorithm 1: Algorithm for dimension reduction with Method 1.

It can easily be shown that

$$\left\| \begin{bmatrix} 0 & v_1^T \Lambda_2 V_2 \\ V_2^T \Lambda_2 v_1 & V_2^T \Lambda_2 V_2 \end{bmatrix} \right\|_F \leq \left\| V^T \Lambda_2 V \right\|_F .$$

By the unitary invariance of the Frobenius norm and the criterion (9), we can use the lemma to prove a bound for the mean square error summarized in the following theorem.

Theorem 1. *Given $\epsilon > 0$ and $\alpha > 0$ such that*

$$\max_{\|a\|_2=1} \max_{\|b\|_2=1} \mathrm{E}[(a^T y)^2 (b^T y)^2] \leq \alpha.$$

Then the $\Delta\Gamma$ approximation Π_1 produced by Algorithm 1 satisfies

$$\mathrm{E}[(\Pi - \Pi_1)^2] \leq \frac{\alpha \epsilon n}{4}.$$

3.2 Method 2

In Section 2, we argued that a $\Delta\Gamma$ approximation (1) can, by a change of variables, be transformed so that the random variables y all have zero mean, are uncorrelated, and have variance one. Then the impact of each of the random variables are approximately equal. A $\Delta\Gamma$ approximation (1) can be written as a quadratic form

$$\Pi = \frac{1}{2}[y^T, 1] \begin{bmatrix} \Gamma & \Delta \\ \Delta^T & 0 \end{bmatrix} \begin{bmatrix} y \\ 1 \end{bmatrix},$$

and, therefore, we can define the distance between two $\Delta\Gamma$ approximations using a matrix norm:

$$\|\Pi - \Pi_1\| = \left\| \begin{bmatrix} \Gamma & \Delta \\ \Delta^T & 0 \end{bmatrix} - \begin{bmatrix} \Gamma_1 & \Delta_1 \\ \Delta_1 & \Xi_1 \end{bmatrix} \right\|_2 . \tag{10}$$

This is the metric that we shall use for dimension reduction.

Consider the Schur decomposition for Π,

$$\begin{bmatrix} \Gamma & \Delta \\ \Delta^T & 0 \end{bmatrix} = U \Lambda U^T,$$

where U is an orthogonal matrix and Λ is a diagonal matrix. The diagonal elements of Λ are the eigenvalues and we may assume that they are ordered in decreasing absolute value $|\lambda_1| \geq \ldots \geq |\lambda_{N+1}|$. Let

$$\Lambda_k = \mathrm{diag}(\lambda_1, \ldots, \lambda_k, 0, \ldots, 0).$$

The Schmidt-Mirsky theorem says that $U \Lambda_k U^T$ solves the minimization problem

$$\min_{\mathrm{rank}(B)=k} \|\Pi - B\|_2 = \|\Pi - U \Lambda_k U^T\|_2 = |\lambda_{k+1}|$$

(for a proof see for example [20, p.208] or [14, p.71]).

Given a tolerance $\epsilon > 0$, the Schmidt-Mirksy gives a tool to find the optimal function Π_1. Let k be the smallest k such that

$$|\lambda_{k+1}| \leq \epsilon.$$

Then,

$$\|\Pi - U\Lambda_k U^T\|_2 \leq \epsilon.$$

The terms of the reduced function Π_1 can be computed from the matrix $U\Lambda_k U^T$. Partition the orthogonal matrix U as

$$U = \begin{bmatrix} U_{11} & U_{12} \\ u_{21}^T & u_{22}^T \end{bmatrix} = \left[\begin{array}{ccc|ccc} u_{11} & \cdots & u_{1k} & u_{1k+1} & \cdots & u_{1n+1} \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ u_{n1} & \cdots & u_{nk} & u_{nk+1} & \cdots & u_{nn+1} \\ \hline u_{n+11} & \cdots & u_{n+1k} & u_{n+1k+1} & \cdots & u_{n+1n+1} \end{array}\right].$$

The Q_1 in (7) can be taken as the $n \times k$ matrix with orthonormal columns in the QR factorization $Q_1 R = U_{11}$. For this choice, we get the reduced $\Delta\Gamma$ approximation

$$\Pi_1 = \frac{1}{2} z_1^T R \Lambda_k R^T z_1 + z_1^T R \Lambda_k u_{21} + u_{21}^T \Lambda_k u_{21}.$$

The steps of Method 2 are summarized in Algorithm 2. Since the constant term cancels, we have chosen not to compute it.

For some matrices, the matrix U_{11} may be rank deficient.[2] This is not a serious problem since it is easy to show that this leads to an approximation of size $(k-1) \times (k-1)$—it is a lucky break that earns us the additional reduction of one dimension.

4 A computational example

In this section, we discuss a computational example. The purpose of this example is twofold. First, we show that dimension reduction effectively reduces computation time and that this can be achieved without sacrificing accuracy. Second, we compare the performance of the two methods presented in the previous section. Method 1 has a computational advantage in that it only needs to solve one eigenvalue problem whereas Method 2 needs to compute an additional singular value decomposition. We show that Method 1 also has a slight accuracy advantage over Method 2 in that it produces

[2] Consider for example

$$\begin{bmatrix} 1 & 0 & \sqrt{2} \\ 0 & 0.1 & 0 \\ \sqrt{2} & 0 & 0 \end{bmatrix} = \left[\begin{array}{cc|c} \frac{\sqrt{2}}{\sqrt{3}} & \frac{1}{\sqrt{3}} & 0 \\ 0 & 0 & 1 \\ \hline \frac{1}{\sqrt{3}} & -\frac{\sqrt{2}}{\sqrt{3}} & 0 \end{array}\right] \left[\begin{array}{cc|c} 2 & 0 & 0 \\ 0 & -1 & 0 \\ \hline 0 & 0 & -0.1 \end{array}\right] \left[\begin{array}{cc|c} \frac{\sqrt{2}}{\sqrt{3}} & \frac{1}{\sqrt{3}} & 0 \\ 0 & 0 & 1 \\ \hline \frac{1}{\sqrt{3}} & -\frac{\sqrt{2}}{\sqrt{3}} & 0 \end{array}\right]^T$$

Input: A mean n-vector m, an $n \times n$ covariance matrix C, an n-vector $\mathbf{\Delta}$, an $n \times n$ matrix Γ, a tolerance $\epsilon > 0$ and a bound for the moments α.

Output: A new $\Delta\Gamma$ approximation Π_1 with fewer dimensions, and the matrix Q_1 that characterizes the transformation of the risk factors.

- Compute the Cholesky decomposition

$$C = R^T R.$$

- Perform the change of variables:

$$\Gamma \leftarrow R\Gamma R^T,$$
$$\Delta \leftarrow R(\Delta + \Gamma m).$$

- Compute the Schur decomposition

$$\begin{bmatrix} \Gamma & \Delta \\ \Delta^T & 0 \end{bmatrix} = U \Lambda U^T.$$

  Order the eigenvalues so that $|\lambda_i| \geq |\lambda_{i+1}|$.

- Find the smallest k such that

$$|\lambda_{k+1}| \leq \epsilon.$$

- Compute the QR factorization

$$Q_1 R = U(1\!:\!n, 1\!:\!k).$$

- Compute Π_1:

$$\Gamma_1 \leftarrow R\Lambda(1\!:\!k, 1\!:\!k)R^T$$
$$\Delta_1 \leftarrow R\Lambda(1\!:\!k, 1\!:\!k)U(n, 1\!:\!k)$$

Algorithm 2: Algorithm for dimension reduction with Method 2.

a more accurate answer when a single dimension is used for the simulation. When more dimensions are used, the two methods produce close to identical estimates for value-at-risk.

For the example, we decided to use a benchmark portfolio of European options. We took a portfolio with one call option and one put option on each of 35 stocks traded on the Toronto Stock Exchange. The stocks are from the Toronto35 index. All options have 3 months to maturity and the strike price is the same as the price of the stock. To get a $\Delta\Gamma$ approximation, we priced the options using the standard Black-Scholes model with 5% interest rate and volatility equal to the empirical volatility estimated from market data (in practice implied volatilities should be used for pricing). It is of course possible to replace the Black-Scholes model with a more sophisticated pricing model. However, we believe that, for the purpose of this experiment, to illustrate the performance of dimension reduction and to compare the two methods, this benchmark portfolio is appropriate.

We estimated the mean and covariance for the return model from daily closing prices over two years, from May 24 1999 to May 24 2001. In this experiment, we used a normal distribution for the risk factors. A model with fat-tails adds little to the issue of dimension reduction and would make the experiment less transparent. We refer to the authors' discussion on how to use non-normal models in the fast convolution method in [1].

To study the effect of dimension reduction, we compute the daily value-at-risk at the 99% confidence level with dimensions $k = 1, \ldots, 35$. The value-at-risk for a long position, an investor holding the options, is \$4.99 CDN and for a short position, an investor selling the options, is \$8.20 CDN. The value of the portfolios is \$243.81 CDN, so the long value-at-risk corresponds to 2.05% of the total portfolio value, and the short value-at-risk corresponds to 3.36%. Figure 1 shows the value-at-risk as a function of the number of dimensions for the two methods. All four plots have two solid lines. The top solid lines is the value-at-risk computed from the reduced portfolio Π_1, i.e., the solution to

$$\mathrm{prob}[\Pi_1 - \mathrm{E}[\Pi_1] < -\mathrm{VaR}\,] = 0.01.$$

The solid line on the bottom is the expectation $\mathrm{E}[\Pi_1]$. Note that the expectation for the short position is the reflection of the expectation for the long position in the horizontal axis. Finally, the dashed line is the absolute value-at-risk, the sum of the value-at-risk and the expectation:

$$\text{absolute value-at-risk} = \mathrm{VaR}\ - \mathrm{E}[\Pi_1].$$

From the figure, we can draw three conclusions. First, Method 1 and Method 2 give similar estimates for value-at-risk, except for the first point, the case when the portfolio is reduced to a problem with only one dimension. When Π_1 has only one dimension, Method 2 does not give a reliable answer. Second, for both methods the reduced portfolio Π_1 gives an estimate close to the actual value-at-risk when a few dimensions are used, the absolute error is less than 1.5% for five dimensions and less than 0.6% for ten.

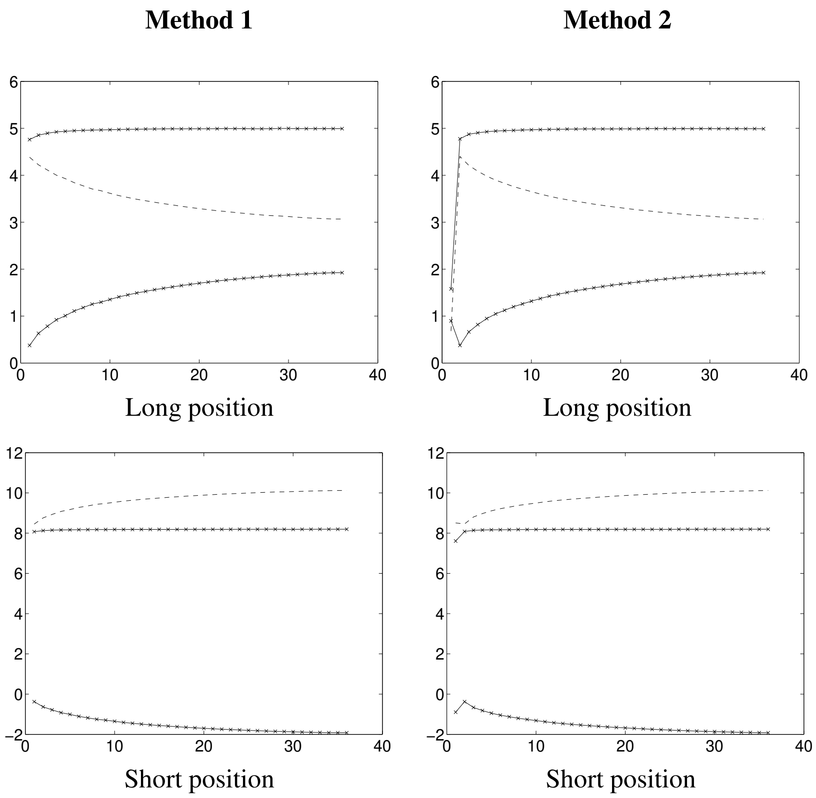


Figure 1: The graphs shows 99% value-at-risk for one day as a function of the number of dimensions used in the computation. The left columns shows the result with Method 1 and the right with Method 2. The first row is for a long position, holding the portfolio, and the second is for a short position, writing or selling the portfolio.

Third, separating the absolute value-at-risk into the value-at-risk and the expected gain is a good idea since the expected gain is easy to compute for the full problem whereas value-at-risk is more cumbersome.

The main incentive for dimension reduction is to reduce the time for computing an estimate for value-at-risk. To illustrate one application where dimension reduction makes a difference, we consider, for the second part of this example, a portfolio with a short position in the option portfolio above and a long position in the stocks that hedges 90% of the Δ for each risk factor. Suppose that we want to hedge this portfolio with a position a in the index and a position b in a call option on the index. The call option is similar to the other options: it has three months to maturity and is at-the-money. Now, we want to solve an optimization problem to find a and b that minimize the value-at-risk. It is our experience that the solution to the optimization problem is more sensitive to changes in the Δ than the Γ and that the absolute value-at-risk is more stable. Therefore, to regularize the problem, we decided to include the expectation and to add a penalty term:

$$\min_{a,b} \mathrm{VaR}\,(a,b) - \mathrm{E}[\Pi] + 0.05|b| \cdot \text{price call}.$$

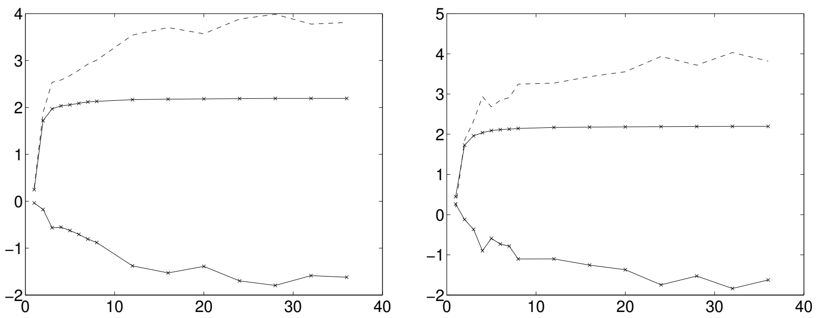
The penalty function expresses a preference for index positions, a reasonable approach since using options means a larger trading cost.

To solve the optimization problem, we used a line search method in MINPACK-1. The results from this optimization is displayed in Figure 2. From these figures we make two observations. First, Method 1 and Method 2 give similar estimates for value-at-risk and lead to similar solutions to the optimization problem. Second, the value-at-risk and hedge parameters, a and b, for the reduced models are accurate when a few dimensions are used. We see that the abrupt changes in the curves for absolute value-at-risk and for the expected return are simultaneous and the jumps offset each other leading to a smooth curve for value-at-risk. Furthermore, the jumps coincide with adjustments in a and b.

Figure 3 shows the time per value-at-risk function call during the optimization.[3] The two methods lead to almost the same time per iteration, i.e., the extra work for solving an additional eigenvalue problem in Method 2 is an insignificant addition to the total time. The figure clearly shows that dimension reduction is an efficient method to limit the computation time. We choose to show the time for average time per function call rather than total optimization time, because we used the output from previous optimization steps as starting guesses for the next problem and therefore a direct comparison of time would not be fair. To get an idea of the savings in total time, we note that for 8 dimensions the optimization took approximately $4\frac{1}{4}$ minutes and all 36 dimensions it took 26 to 28 minutes, a significant savings in problem time.

[3]The figure shows the median average times for each dimension over three repetitions of the simulation.

Method 1 **Method 2**



# of dim.	Index	Call
1	-0.8589	1.7668
2	-0.8584	1.7670
3	-0.8396	1.7137
4	-0.8396	1.7137
5	-0.8396	1.7143
6	-0.8396	1.7143
7	-0.8402	1.7102
8	-0.8362	1.7105
12	-0.8362	1.7105
16	-0.8362	1.7105
20	-0.8361	1.7104
24	-0.8290	1.6823
28	-0.8290	1.6823
32	-0.8290	1.6823
36	-0.8290	1.6823

# of dim.	Index	Call
1	-0.7993	1.8002
2	-0.8376	1.7473
3	-0.8388	1.7395
4	-0.8388	1.7395
5	-0.8146	1.6790
6	-0.8146	1.6790
7	-0.8146	1.6790
8	-0.8146	1.6790
12	-0.8146	1.6793
16	-0.8147	1.6619
20	-0.8145	1.6607
24	-0.8156	1.6562
28	-0.8102	1.6537
32	-0.8102	1.6537
36	-0.8116	1.6535

Figure 2: Hedging an option portfolio with an index call option and a position in the index. The graphs shows value-at-risk for the hedged portfolio as a function of the number of risk factors used. The lines are as before.

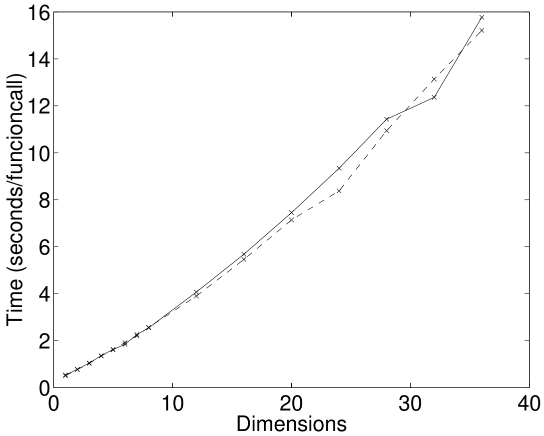


Figure 3: Time in seconds per value-at-risk function call. The time for Method 1 is the solid line and the time for Method 2 is the dashed line. The times are mean average time per function call for three simulations for the same problem.

5 Conclusions

In this paper, we have presented two new methods for dimension reduction in value-at-risk algorithms. The purpose of dimension reduction is to find a portfolio that depends on fewer dimensions, that still has the main properties of the original one, and thereby to reduce the computation time. Method 1 computes a new $\Delta\Gamma$ approximation that is close in the sense that the mean square error is small. Method 2 uses the identification of quadratic forms with matrices and finds a reduced $\Delta\Gamma$ approximation by computing a nearby matrix of low rank. To illustrate the usefulness of dimension reduction and to compare the two methods, we studied a portfolio of options on the stocks in the Toronto35 index. The example shows that both methods lead to similar results and that a few dimensions is enough to get an accurate estimate to value-at-risk. Finally, we show an example of dimension reduction in an optimization problem for hedging an option portfolio. In optimization, value-at-risk needs to be computed many times and, for problems with many dimensions, dimension reduction leads to large savings in computation time.

References

[1] C. Albanese, K. Jackson, and P. Wiberg. Hedging with value-at-risk. Technical report, University of Toronto, 2001.

[2] Basle Committee on Banking Supervision. International convergence of capital measurement and capital standards. Technical report, Bank for International Settlements, 1988.

[3] Basle Committee on Banking Supervision. Amendment to the Capital Accord to incorporate market risks. Technical report, Bank for International Settlements, 1996.

[4] Basle Committee on Banking Supervision. The new Basel Capital Accord. Technical report, Bank for International Settlements, 1996.

[5] T.S. Beder. VAR: Seductive but dangerous. *Financial Analysts Journal*, 51(5):12–24, 1995.

[6] M. Britten-Jones and S.M. Schaefer. Non-linear value-at-risk. *European Finance Review*, 2(2):161–187, 1998.

[7] J.Y. Campbell, A.W. Lo, and A.C. MacKinlay. *The Econometrics of Financial Markets*. Princeton University Press, 1996.

[8] D. Duffie and J. Pan. An overview of Value at Risk. *Journal of Derivatives*, 4(3):7–49, 1997.

[9] D. Duffie and J. Pan. Analytic value-at-risk with jumps and credit risk. *Finance and Stochastics*, 5(2):155–180, 2001.

[10] L. El-Jahel, W. Perraudin, and P. Sellin. Value at risk for derivatives. *Journal of Derivatives*, 6(3):7–26, 1999.

[11] P. Embrechts, A. McNeil, and D. Straumann. Correlation and dependency in risk management: properties and pitfalls. Technical report, Dept. of Mathematics, ETH, 1999.

[12] P. Embrechts, A. McNeil, and D. Straumann. Correlation: Pitfalls and alternatives. *RISK*, 12(4), May 1999.

[13] A. Feuerverger and A.C.M. Wong. Computation of value-at-risk for non-linear portfolios. *Journal of Risk*, 3(3):37–55, Fall 2000.

[14] G.H. Golub and C.F. Van Loan. *Matrix Computations*. The Johns Hopkins University Press, Baltimore, USA, second edition, 1989.

[15] J.C. Hull. *Options, Futures and Other Derivatives*. Prentice Hall, 4th edition edition, 2000.

[16] A. Kreinin, L. Merkoulovitch, D. Rosen, and M. Zerbs. Principal component analysis in quasi monte carlo simulation. *Algo Research Quarterly*, 1(2):21–30, December 1998.

[17] J. Mina and A. Ulmer. Delta-Gamma four ways. Technical report, RiskMetrics Group, 1999.

[18] J.P. Morgan/Reuters. *RiskMetrics—Technical Document*. J.P. Morgan, 4th edition edition, 1996.

[19] M. Reimers and M. Zerbs. Dimension reduction by asset blocks. *Algo Research Quarterly*, 1(2):43–58, December 1998.

[20] G.W. Stewart and J.-G. Sun. *Matrix Perturbation Theory*. Academic Press, London, UK, 1990.

[21] Paul Wilmott. *Paul Wilmott on Quantitative Finance*. John Wiley & Sons, 2000.

[22] www.gloriamundi.com. Web site devoted to value-at-risk.