Restoring Definiteness via Shrinking, with an Application to Correlation Matrices with a Fixed Block

Higham, Nicholas J. and Strabic, Natasa and Sego, Vedran

2014

MIMS EPrint: **2014.54**

ISSN 1749-9097

RESTORING DEFINITENESS VIA SHRINKING, WITH AN APPLICATION TO CORRELATION MATRICES WITH A FIXED BLOCK[*]

NICHOLAS J. HIGHAM[†], NATAŠA STRABIĆ[†], AND VEDRAN ŠEGO[‡]

Abstract. Indefinite estimates of positive semidefinite matrices arise in many data analysis applications involving covariance matrices and correlation matrices. We develop a method for restoring positive semidefiniteness of an indefinite estimate based on the process of shrinking, which finds a convex linear combination $S(\alpha) = \alpha M_1 + (1-\alpha)M_0$ of the original matrix M_0 and a target positive semidefinite matrix M_1. We describe three algorithms for computing the optimal shrinking parameter $\alpha_* = \min\{\alpha \in [0,1] : S(\alpha) \text{ is positive semidefinite}\}$. One algorithm is based on the bisection method, with the use of Cholesky factorization to test definiteness, a second employs Newton's method, and a third finds the smallest eigenvalue of a symmetric definite generalized eigenvalue problem. We show that weights that reflect confidence in the individual entries of M_0 can be used to construct a natural choice of the target matrix M_1. We treat in detail a problem variant in which a positive semidefinite leading principal submatrix of M_0 remains fixed, showing how the fixed block can be exploited to reduce the cost of the bisection and generalized eigenvalue methods. Numerical experiments show that when applied to estimates of correlation matrices shrinking can be at least an order of magnitude faster than computing the nearest correlation matrix.

Key words. covariance matrix, correlation matrix, indefinite matrix, positive semidefinite matrix, shrinking, bisection method, Newton's method, symmetric definite generalized eigenvalue problem, weighting, fixed block

AMS subject classifications. 15A57, 65F30

1. Introduction.

Covariance matrices and correlation matrices estimated from discrete sets of empirical data play a key role in many applications. These matrices are symmetric positive semidefinite, with a correlation matrix also having unit diagonal. However, in practice the estimated matrices may lack definiteness due to missing or asynchronous observations or by the nature of their construction, such as in stress testing in finance or when correlations are determined within groups and then the groups are joined together. To ensure the validity of the subsequent analysis the invalid estimate needs to be replaced by a valid covariance or correlation matrix. Such a replacement is needed in a very wide variety of applications, of which some recent examples include modelling public health [7] and dietary intakes [37], analysis of wind power generators [10], determination of insurance premiums for crops [12], reservoir modelling [26], oceanography [33], and horse breeding [36].

Approaches to repairing an invalid covariance matrix include computing the nearest positive semidefinite matrix in the Frobenius norm, which amounts to shifting all the negative eigenvalues to zero while keeping the eigenvectors fixed [15], and ridge regression, in which a multiple of the identity is added to the matrix to make it positive semidefinite. For repairing invalid correlation matrices a simple approach is to compute the nearest positive semidefinite matrix and then to diagonally scale it to

[*] Version of November 17, 2014. The work of the first two authors was supported by European Research Council Advanced Grant MATFUN (267526). The first author was also supported by Engineering and Physical Sciences Research Council grants EP/I03112X/1 and EP/I01912X/1.

[†] School of Mathematics, The University of Manchester, Manchester, M13 9PL, UK (nick.higham@manchester.ac.uk, http://www.maths.manchester.ac.uk/~higham, natasa.strabic@manchester.ac.uk, http://www.maths.manchester.ac.uk/~nstrabic)

[‡] School of Mathematics, The University of Manchester, Manchester, M13 9PL, UK and The University of Zagreb, Department of Mathematics, Bijenička cesta 30, 10 000 Zagreb, Croatia (vsego@math.hr, vsego@vsego.org)

restore the unit diagonal. As with ridge regression, this approach may change the matrix more than necessary. A widely used alternative is to compute the nearest correlation matrix to the original matrix in the Frobenius norm [4], [18], [30].

In this paper we propose a new method to repair invalid covariance and correlation matrices, inspired by an idea from statistics called *shrinking*. The original idea of shrinking is to compute a convex linear combination (a weighted average) of two positive semidefinite matrices M_0 and M_1, where M_0 is a given matrix whose performance we wish to improve and M_1 is a target matrix that is appropriately chosen for each specific application. Shrinking incorporates the idea of "borrowing strength across the variables", with the hope that the shrinkage estimator $\alpha M_1 + (1-\alpha)M_0$, where $\alpha \in [0,1]$ is the chosen shrinking parameter, inherits the good properties of the target and is thus more suitable for use than either of the extremes.

Shrinking has been successfully applied to improving the properties of both valid covariance matrices [22], [23], [24], [35] and valid correlation matrices [29]. Different choices of the target matrix have been considered in the literature, including multiples of the identity matrix, diagonal matrices, and the matrix of all ones; see [35, Table 2] for a summary. The central part of the method is selecting the shrinkage parameter α. In some cases it is fixed at a certain value or made to depend on the sample size. A data-driven objective to compute α by minimizing a loss function is proposed in [22] and extended in [35].

While the original idea of shrinking was to compute a convex linear combination of two positive semidefinite matrices, we use it in a different context: as a means of transforming an *indefinite* covariance or correlation matrix approximation so that it becomes positive semidefinite. The possibility of using shrinking for this purpose was mentioned by Devlin, Gnanadesikan, and Kettenring [9, sec. 4.4] and also by Kupiec [20, sec. 5], who suggests a "trial and error" way of choosing the shrinkage parameter α. Rebonato and Jäckel [34] criticize Kupiec's suggestion on the grounds that it is expensive, since each trial requires a full eigenvalue decomposition, that a target matrix must be chosen, and that "there is no way of determining to what extent the resulting matrix is optimal in any easily quantifiable sense."

We present a shrinking method that overcomes the drawbacks pointed out by Rebonato and Jäckel. We define an optimal shrinking parameter that produces a minimal elementwise perturbation in a direction determined by the target matrix. We propose three algorithms for computing the optimal parameter, none of which requires repeated full eigenvalue decompositions. We also show how a target matrix can be constructed in a natural way based on the confidence with which the elements of the given matrix are known.

Often, additional requirements are put on a replacement matrix, and in this paper we focus on the case where a positive semidefinite leading principal submatrix of an approximate correlation matrix is known to be exact and needs to be preserved. To illustrate the importance of this specific problem variant, we next present two applications where this requirement occurs: dealing with incomplete data sets and correlation stress testing.

To see how keeping a block fixed naturally appears as a condition in dealing with incomplete data sets, suppose that the data from K observations of N random variables is collected in a matrix $X \in \mathbb{R}^{K\times N}$ and that some of the entries are missing. Without loss of generality, we can permute the columns of X so that there are no missing elements in the first m columns. The pairwise deletion method [25, sec. 2.2], which in calculating the correlation between a pair of vectors uses only the components available in both vectors, leads to an approximate sample correlation matrix of the form

$$M_0 = \begin{array}{c c} & \begin{array}{cc} m & n \end{array} \\ \begin{array}{c} m \\ n \end{array} & \begin{bmatrix} A & Y \\ Y^T & B \end{bmatrix} \end{array} \in \mathbb{R}^{N\times N},$$

where $A \in \mathbb{R}^{m\times m}$ is a valid correlation matrix since it contains the exact correlations between variables $1{:}\, m$, which have no missing observations. In contrast, $Y \in \mathbb{R}^{m\times n}$ and $B \in \mathbb{R}^{n\times n}$, with $m + n = N$, contain approximate correlations because their elements are computed from an incomplete data set. The matrix M_0 is symmetric with unit diagonal and off-diagonal elements in $[-1, 1]$, but there is no guarantee that it is positive semidefinite. Our goal is to modify M_0 to make it positive semidefinite while preserving the unit diagonal and the $(1, 1)$ block A. We take as the target the matrix $M_1 = \mathrm{diag}(A, I)$, for which the optimal α can be interpreted as the minimal relative change applied uniformly to all unfixed elements of M_0.

Stress testing is a method used in finance to explore the impact on a financial instrument of pushing risk parameters toward more extreme levels. In correlation stress testing [11], [32] we have a valid correlation matrix

$$C = \begin{bmatrix} C_{11} & C_{12} \\ C_{12}^T & C_{22} \end{bmatrix} \in \mathbb{R}^{N\times N},$$

where C_{11} and C_{22} are correlation matrices corresponding to a first and second group of assets and C_{12} is the cross-group correlation matrix, and wish to modify it by replacing C_{22} with a new correlation matrix $\widehat{C}_{22}$ that reflects changes to the second group of assets. If the modified matrix is indefinite it needs to be replaced with a valid correlation matrix, but the C_{11} block should remain unchanged since the first group of assets was not affected. Again, shrinking can be applied, with $M_1 = \mathrm{diag}(C_{11}, I)$.

An application which generalizes keeping just one diagonal block fixed is risk aggregation [1], [19]. Here, diagonal blocks of a global covariance or correlation matrix represent individual markets and these can be individually updated with more refined local forecasts. Replacing the old diagonal blocks with the new ones can result in the global matrix becoming indefinite, so the covariances or correlations in the off-diagonal blocks have to be readjusted while keeping the new diagonal blocks unchanged. Shrinking can be used to solve this problem by selecting as target the matrix comprised of the new diagonal blocks and the optimal α can again be interpreted as the minimal relative change applied uniformly to all unfixed elements of the global matrix.

The paper is organized as follows. In the next section we define the shrinking problem, characterize the solution, and discuss the choice of target matrix. We present three methods to compute the optimal shrinking parameter in section 3: one based on the bisection method, a second based on Newton's method, and a third that solves a symmetric definite generalized eigenvalue problem. In section 4 we explain how weights can be incorporated into the choice of a target matrix. In section 5 we focus on restoring the definiteness of a correlation matrix while preserving a specified positive semidefinite leading principal submatrix. We show how the bisection and generalized eigenvalue methods can be adapted to exploit the problem structure, explain how the case of a singular fixed block can be reduced to the nonsingular case, show how a lower bound on the smallest eigenvalue can be incorporated, and describe how multiple fixed diagonal blocks can be treated. Numerical experiments are presented in section 6 and concluding remarks are given in section 7.

2. The shrinking problem. Given a real symmetric indefinite matrix M_0 of order N our task is to modify M_0 to make it positive semidefinite by computing a convex linear combination of M_0 and a chosen positive semidefinite target matrix M_1. Hence we consider the matrix

$$S(\alpha) = \alpha M_1 + (1-\alpha)M_0, \quad \alpha \in [0,1]. \tag{2.1}$$

Clearly, $S(\alpha)$ is symmetric for every α, $S(0) = M_0$ is indefinite, and $S(1) = M_1$ is positive semidefinite. We define the optimal shrinking parameter as

$$\alpha_* = \min\{\, \alpha \in [0,1] \colon S(\alpha) \text{ is positive semidefinite} \,\}. \tag{2.2}$$

Since $S(\alpha) = M_0 + \alpha(M_1 - M_0)$, it is clear that we are seeking the elementwise minimal change to M_0 in the direction $M_1 - M_0$.

For another interpretation, note that $M_0 - S(\alpha) = \alpha(M_0 - M_1)$, so $\|M_0 - S(\alpha)\| = \alpha\|M_0 - M_1\|$. Since $M_0 - M_1$ is fixed, this means that $S(\alpha_*)$ is the nearest positive semidefinite matrix to M_0 of the form $S(\alpha)$, in any norm.

We now characterize the optimal shrinking parameter α_*. The following results form the basis for the bisection and Newton methods for computing α_* proposed in sections 3.1 and 3.2.

Since a symmetric matrix is positive semidefinite if and only if its smallest eigenvalue is nonnegative, we focus on the function $f \colon \mathbb{R} \to \mathbb{R}$ defined by

$$f(\alpha) = \lambda_{\min}\left(S(\alpha)\right), \tag{2.3}$$

where $\lambda_{\min}$ denotes the smallest eigenvalue of a symmetric matrix. Note that f is a continuous function, since the eigenvalues of a matrix are continuous functions of its elements. Hence α_* is characterized as

$$\alpha_* = \min\{\, \alpha \in [0,1] \colon f(\alpha) \geq 0 \,\}.$$

Recall that a function $g \colon \mathbb{R} \to \mathbb{R}$ is concave if for every $\alpha_1, \alpha_2 \in \mathbb{R}$ and $t \in [0,1]$,

$$g\bigl(t\alpha_1 + (1-t)\alpha_2\bigr) \geq t g(\alpha_1) + (1-t) g(\alpha_2).$$

In the following lemma we show that the function f defined in (2.3) is concave. In the proof below we will use the characterization, for symmetric C,

$$\lambda_{\min}(C) = \min\{\, x^T C x \colon x^T x = 1 \,\}. \tag{2.4}$$

Lemma 2.1. *The function f in (2.3) is concave on $\mathbb{R}$.*

Proof. Let $\alpha_1, \alpha_2 \in \mathbb{R}$ and $t \in [0,1]$ be arbitrary and note that $S(\alpha)$ is an affine function of α. Then we have

$$\begin{aligned} f(t\alpha_1 + (1-t)\alpha_2) &= \lambda_{\min}\bigl(S(t\alpha_1 + (1-t)\alpha_2)\bigr) \\ &= \lambda_{\min}\bigl(tS(\alpha_1) + (1-t)S(\alpha_2)\bigr) \\ &\geq \lambda_{\min}\bigl(tS(\alpha_1)\bigr) + \lambda_{\min}\bigl((1-t)S(\alpha_2)\bigr) \qquad \text{by (2.4)} \\ &= t\lambda_{\min}\bigl(S(\alpha_1)\bigr) + (1-t)\lambda_{\min}\bigl(S(\alpha_2)\bigr) \\ &= tf(\alpha_1) + (1-t)f(\alpha_2). \qquad \square \end{aligned}$$

Since $f(0) < 0$, $f(1) = \lambda_{\min}(S(1)) = \lambda_{\min}(M_1)$ and f is concave and continuous, it follows that α_* is the unique zero of f in $[0,1]$ if the matrix M_1 is positive definite. In principle we need to allow M_1 to be positive semidefinite and singular, as can happen in our correlation matrix application discussed in section 5, but as we show in section 5.4 in that case the problem can be reduced to the case in which M_1 is positive definite.

When the only goal is to repair the indefiniteness of the matrix M_0, the target matrix M_1 can be chosen as any positive semidefinite matrix. When M_0 is an invalid correlation matrix, which we take to mean that it is indefinite but has unit diagonal, and we want $S(\alpha_*)$ to be a valid correlation matrix, then from (2.1) it follows that the target matrix needs to be a correlation matrix, so that the unit diagonal is preserved. Hence, the simplest target in this case is the identity matrix.

In a setting with time-varying matrices a natural target is the (shrunk) matrix from the previous time step.

3. Computing the optimal shrinking parameter.

We present three methods to compute the optimal shrinking parameter α_* when M_0 is symmetric indefinite and the target matrix M_1 is positive definite. Recall that in this case α_* is the unique zero in $[0,1]$ of f in (2.3).

3.1. Bisection method.

The simplest iterative method to find a zero of a function on a given interval is the bisection method, which yields the following algorithm for our problem.

ALGORITHM 3.1 (Bisection method). *Given the indefinite matrix $M_0 \in \mathbb{R}^{N\times N}$, a positive definite target matrix $M_1 \in \mathbb{R}^{N\times N}$, and a convergence tolerance* tol *this algorithm uses the bisection method to compute the optimal shrinking parameter α_* defined by* (2.2).

```
 1  $\alpha_\ell = 0$, $\alpha_r = 1$
 2  while $\alpha_r - \alpha_\ell >$ tol
 3        $\alpha_m = (\alpha_\ell + \alpha_r)/2$
 4        if $S(\alpha_m)$ is not positive semidefinite
 5            $\alpha_\ell = \alpha_m$
 6        else
 7            $\alpha_r = \alpha_m$
 8        end
 9  end
10  $\alpha_* = \alpha_r$.
```

In the last line of the algorithm we have set α_* to α_r rather than to the generally more accurate value $(\alpha_\ell + \alpha_r)/2$ in order to ensure that $S(\alpha_*)$ is positive semidefinite.

The main computational task in Algorithm 3.1 is testing for positive semidefiniteness. As argued in [15, sec. 5], an arbitrarily small perturbation can make a singular positive semidefinite matrix become positive definite and hence in finite precision arithmetic testing for positive semidefiniteness is equivalent to testing for positive definiteness. One way to test for definiteness is to compute the eigenvalues of $S(\alpha_m)$ and check whether the smallest one is positive. A less expensive approach is to attempt to compute the Cholesky factorization of $S(\alpha_m)$ and declare the matrix positive definite if the process succeeds. Although it might seem numerically dangerous to apply Cholesky factorization to a potentially indefinite matrix, this approach is numerically stable [15, sec. 5]. Hence, we replace step 4 in Algorithm 3.1 with "if the Cholesky factorization of $S(\alpha_m)$ breaks down".

The number of steps needed by Algorithm 3.1 is $\lceil |\log_2 \mathrm{tol}|\rceil$, where the ceiling function $\lceil \alpha \rceil$ denotes the smallest integer greater than or equal to α, reflecting the linear convergence of the bisection method. However, in practical applications the data is often accurate only to three or four significant digits, in which case the tolerance tol will be of order 10^{-4} or larger and bisection will need less than 15 iterations. The cost per step of Algorithm 3.1 depends on the number of successful elimination stages in the Cholesky factorization of $S(\alpha_m)$ and is at most $N^3/3$ flops.

3.2. Newton's method. For a method with faster convergence than the bisection method it is natural to turn to Newton's method, defined by $\alpha_{k+1} = \alpha_k - f(\alpha_k)/f'(\alpha_k)$. If $\lambda_{\min}(S(\alpha))$ is a simple eigenvalue then f in (2.3) is differentiable and [13, sec. 7.2.2]

$$f'(\alpha) = x(\alpha)^T S'(\alpha) x(\alpha) \neq 0, \tag{3.1}$$

where $x(\alpha)$ is a unit norm eigenvector for $\lambda_{\min}(S(\alpha))$ and, from (2.1), $S'(\alpha) = M_1 - M_0$. Note that $S'(\alpha)$ is independent of α.

Lemma 3.2. *The Newton iteration for finding a zero of $f(\alpha) = \lambda_{\min}(S(\alpha))$, where $S(\alpha)$ is defined in (2.1), can be written as*

$$\alpha_{k+1} = \frac{x(\alpha_k)^T M_0 \, x(\alpha_k)}{x(\alpha_k)^T \left(M_0 - M_1\right) x(\alpha_k)}, \tag{3.2}$$

where $x(\alpha_k)$ is a unit norm eigenvector for $\lambda_{\min}(S(\alpha_k))$ and it is assumed that $\lambda_{\min}(S(\alpha_k))$ is simple for each k.

Proof. Dropping the index k for simplicity, let us look at the quotient $f(\alpha)/f'(\alpha)$. We have, from (2.1),

$$f(\alpha) = \lambda_{\min}(S(\alpha)) = x(\alpha)^T S(\alpha) x(\alpha) = x(\alpha)^T \left(M_0 + \alpha(M_1 - M_0)\right) x(\alpha)$$

and, from (3.1),

$$f'(\alpha) = x(\alpha)^T S'(\alpha) x(\alpha) = x(\alpha)^T \left(M_1 - M_0\right) x(\alpha).$$

Hence

$$\alpha_{k+1} = \alpha_k - f(\alpha_k)/f'(\alpha_k) = \alpha_k - \alpha_k - \frac{x(\alpha_k)^T M_0 \, x(\alpha_k)}{x(\alpha_k)^T (M_1 - M_0) x(\alpha_k)},$$

which yields the result. □

Recall that we are looking for $\alpha_* \in [0,1]$ such that $f(\alpha_*) = 0$, where f is continuous and concave with $f(0) < 0$ and $f(1) > 0$. The function f is monotone increasing on an interval $[0, \beta] \subseteq [0,1]$ that contains α_*, but β is not necessarily equal to 1 (and it is possible that f decreases on $[\beta, 1]$). Therefore we have the geometrically obvious result that for any $\alpha_0 < \alpha_*$ the Newton iterates converge monotonically to α_* and hence the Newton method for our problem is globally convergent. In practice, we can set $\alpha_0 = 0$.

Taking all of this into consideration, we have the following algorithm.

Algorithm 3.3 (Newton method). *Given the indefinite matrix $M_0 \in \mathbb{R}^{N \times N}$, a positive definite target matrix $M_1 \in \mathbb{R}^{N \times N}$, and a convergence tolerance* tol *this algorithm uses Newton's method to compute the optimal shrinking parameter α_* defined by (2.2).*

```
$\alpha_0 = 0,\ k = 0$
while not converged to within tolerance tol
    Compute $x(\alpha_k)$, a unit norm eigenvector for $\lambda_{\min}(S(\alpha_k))$
    by tridiagonalization followed by bisection and inverse iteration.
    Compute the new iterate $\alpha_{k+1}$ by (3.2).
    $k = k + 1$
end
$\alpha_* = \alpha_k$
```

A possible stopping test is $|\alpha_{k+1} - \alpha_k| \leq \text{tol}$, which corresponds to the bisection stopping criterion.

The main computational work in the algorithm is computing a unit norm eigenvector for the smallest eigenvalue at each step, and of the many methods that compute one or a few of the (extremal) eigenvalues and their corresponding eigenvectors we have chosen tridiagonalization followed by the bisection method and inverse iteration [8, sec. 5.3.4]. Other possibilities include the power method [13, sec. 8.2.1], orthogonal iteration [13, sec. 8.2.4] and the Lanczos method [13, sec. 10.1].

Note that there is no guarantee that the computed α_* from Algorithm 3.3 will in fact define a positive semidefinite $S(\alpha_*)$ since the iterates stay to the left of α_*.

3.3. Generalized eigenvalue problem. The third method for computing the optimal shrinking parameter is essentially different from the root-finding methods presented above and provides the most elegant description of α_*. Recall that we are looking for the smallest $\alpha \in [0, 1]$ for which the matrix

$$S(\alpha) = \alpha M_1 + (1 - \alpha) M_0 =: E - \alpha F, \tag{3.3}$$

is positive semidefinite. The matrix $S(\alpha)$ is a symmetric matrix for every α, which means that it has real eigenvalues. Then $\alpha \mapsto \lambda_1(S(\alpha)), \ldots, \alpha \mapsto \lambda_N(S(\alpha))$ is a continuous parametrization of the N eigenvalue functions $\lambda_1 \geq \cdots \geq \lambda_N$, and in this notation, $\lambda_N = f$ in (2.3).

If α is such that $\lambda_k(S(\alpha)) = 0$ for some k then the matrix $S(\alpha)$ is singular which means, by definition, that α is a generalized eigenvalue of the pencil $E - \alpha F$. It follows that α_*, the zero of λ_N, is a generalized eigenvalue of the matrix pencil $E - \alpha F$, and among all generalized eigenvalues in $[0, 1]$, α_* is the rightmost one.

The matrices $E = M_1$ and $F = M_0 - M_1$ are symmetric indefinite, and the QZ algorithm for computing the generalized eigenvalues of $E - \alpha F$ cannot exploit the symmetry. However, some symmetric pencils, known as definite pencils, can be transformed into pencils in which one of the matrices is positive definite, and there are algorithms for attempting to find such a transformation [5], [14]. In our case, it is trivial to obtain a definite pencil. We write

$$S(\alpha) = (1 - \alpha)\left(\frac{\alpha}{1 - \alpha} M_1 + M_0\right)$$

and, since $\alpha_* < 1$, $S(\alpha)$ is singular precisely at the generalized eigenvalues of the definite pencil $M_0 - \mu M_1$, where $\mu = \alpha/(\alpha - 1)$. We find α_* by computing the smallest generalized eigenvalue of this pencil. To do so we transform it to a standard symmetric eigenvalue problem $C - \mu I$, where $C = R^{-T} M_0 R^{-1}$ and $M_1 = R^T R$ the Cholesky factorization of M_1; see, for example, [6]. To compute the smallest eigenvalue of the matrix C we use tridiagonalization followed by the bisection method.

Table 3.1

Approximate costs, in flops, of k_1 iterations of the bisection algorithm (Algorithm 3.1 with Cholesky factorization), k_2 iterations of Newton's method (Algorithm 3.3), and the generalized eigenvalue-based algorithm (Algorithm 3.4), all for M_0 of size N.

Bisection	Newton	Generalized eigenvalue
$\dfrac{k_1N^3}{3}$	$\dfrac{4k_2N^3}{3}$	$3N^3$

The algorithm can be summarized as follows.

Algorithm 3.4 (Generalized eigenvalue method). *Given the indefinite matrix $M_0 \in \mathbb{R}^{N\times N}$ and a positive definite target matrix $M_1 \in \mathbb{R}^{N\times N}$, this algorithm uses the generalized eigenvalue interpretation to compute the optimal shrinking parameter α_* defined by (2.2).*

```
1 Compute the Cholesky factorization $M_1 = R^TR$.
2 Form $C = R^{-T}M_0R^{-1}$ by multiple right-hand side triangular solves.
3 Find $\mu_*$, the smallest eigenvalue of $C$, by tridiagonalization followed by
  bisection.
4 $\alpha_* = \mu_*/(\mu_* - 1)$.
```

3.4. Comparison. Table 3.1 summarizes the approximate costs in flops of the three algorithms. Which method is the cheapest depends on the desired accuracy. The bisection algorithm has the advantage of being the easiest to implement and it guarantees a positive semidefinite solution. Moreover, it also can exploit a relatively large value of tol (corresponding to low precision data) better than the other two methods, for which the cost of the iterative bisection and inverse iteration phase is negligible compared with the cost of the preceding computations with full matrices.

Note that when $M_1 = I$, the first two lines of Algorithm 3.4 are empty and the cost reduces to $4N^3/3$ flops.

4. Introducing weights.

In practice, different elements of M_0 may be known to different levels of accuracy or confidence [2]. This can be reflected by introducing a symmetric matrix $W \in \mathbb{R}^{N\times N}$ of nonnegative weights $w_{ij} \in [0,1]$ and defining the target matrix as $M_1 = W \circ M_0$, where $\circ$ is the Hadamard (elementwise) product. Then

$$S(\alpha)_{ij} = \big(1 + \alpha(w_{ij} - 1)\big)(M_0)_{ij}.$$

Therefore a weight $w_{ij} = 1$ signifies that the (i,j) element of M_0 must not be changed, while a weight $w_{ij} = 0$ allows that element to be changed as much as necessary. Intermediate values $w_{ij} \in (0,1)$ put a greater restriction (for larger w_{ij}) or lesser restriction (for smaller w_{ij}) on the relative amount by which the (i,j) elements of M_0 can change.

The unit diagonal in correlation problems poses no difficulties as it is simply obtained for W with a unit diagonal. However, note that there is no guarantee that M_1 obtained this way is positive semidefinite, which is a requirement for a target matrix in the shrinking method. If the target matrix turns out to be indefinite then the weights are too restrictive and W should be modified.

Note that weighting provides a natural answer to the question of how to choose the target matrix: the target is based on the original information in M_0 and the trust that can be put in each individual entry.

Since weighting is reflected entirely in the target matrix M_1, all the methods from section 3 apply without change. This is in contrast to the nearest correlation matrix problem in the Frobenius norm, where the so-called H-weighted problem $\min\{\, \|W \circ (M_0 - X)\|_F^2 : X \text{ is a correlation matrix} \,\}$ is much more challenging to solve than the unweighted problem with $w_{ij} = 1$ [18], [31]. For example, the NAG code `g02aaj/nag_nearest_correlation_h_weight` solves the H-weighted nearest correlation matrix problem, but the documentation says that if the weights vary by several orders of magnitude then the underlying algorithm may fail to converge [28].

5. Correlation matrix with fixed block.

We now consider an important special case of weighting in which the given matrix M_0 is an invalid correlation matrix and has a positive semidefinite leading principal submatrix that must remain fixed. As explained in section 1 this problem arises when a correlation matrix is formed from incomplete data sets or through stress testing.

One approach is to replace M_0 by the nearest correlation matrix in the Frobenius norm with a prescribed block fixed, which can be computed by a simple modification of the alternating projections method from [18]. The modified algorithm and numerical experiments with it can be found in [3], [25]. The algorithm is guaranteed to converge to the unique solution of the problem but its convergence can be slow.

We derive a new alternative based on shrinking. Recall from section 1 that we have

$$M_0 = \begin{array}{cc} & \begin{array}{cc} m & n \end{array} \\ \begin{array}{c} m \\ n \end{array} & \begin{bmatrix} A & Y \\ Y^T & B \end{bmatrix} \end{array} \in \mathbb{R}^{N\times N}, \quad A \text{ a correlation matrix}, \quad b_{ii} = 1,\ i = 1\colon n, \tag{5.1}$$

and we wish for A and the unit diagonal of B to remain unchanged. Hence, the $(1,1)$ block of the target matrix M_1 must equal A and the $(2,2)$ block must have a unit diagonal. The target matrix

$$M_1 = \operatorname{diag}(A, I) \tag{5.2}$$

is the simplest matrix that satisfies these conditions. We are looking for

$$\alpha_* = \min\{\, \alpha \in [0,1] \colon f(\alpha) \ge 0 \,\}, \tag{5.3}$$

with $f(\alpha) = \lambda_{\min}(S(\alpha))$ and

$$S(\alpha) = \begin{bmatrix} A & (1-\alpha)Y \\ (1-\alpha)Y^T & \alpha I + (1-\alpha)B \end{bmatrix}. \tag{5.4}$$

In addition to the interpretation mentioned in section 2 that the resulting matrix $S(\alpha_*)$ is the elementwise minimal change of M_0 in the direction $M_1 - M_0$, here we also have that α_* is the minimal relative change applied uniformly to all the unfixed elements of M_0.

A desirable property is that if the rows and columns of A and of B are symmetrically permuted then $S(\alpha_*)$ is permuted in the same way. It is easy to show that this is the case, using the formulae in section 5.4.

We first assume that A is positive definite, so that we have a positive definite target matrix. The matrix $S(\alpha)$ has special structure: its leading positive definite block A does not change with α and this can be very efficiently exploited in the bisection method (Algorithm 3.1) and the generalized eigenvalue approach (Algorithm 3.4) to compute α_*, as we show in the next two sections.

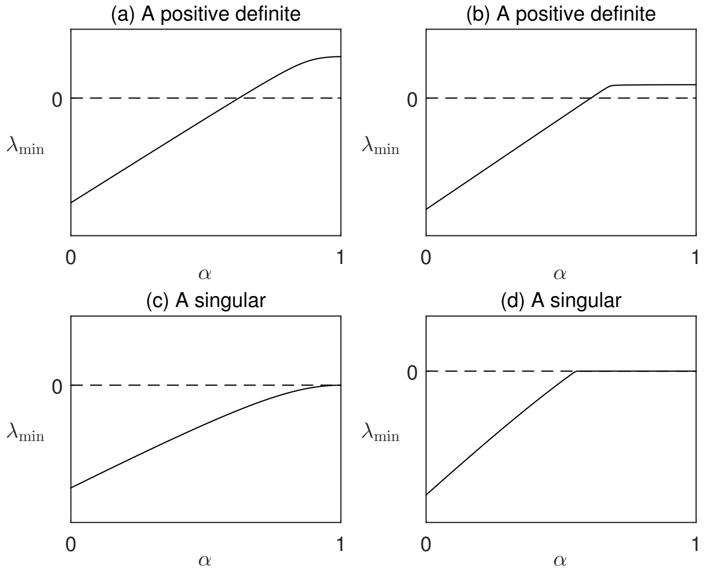

FIG. 5.1. *Plots of the function $f(\alpha) = \lambda_{\min}(S(\alpha))$ for $S(\alpha)$ in (5.4) for A positive definite in (a) and (b) and positive semidefinite and singular in (c) and (d).*

Having the fixed block A singular leads to significant changes to both the problem and the proposed methods for computing the optimal shrinking parameter, as discussed further in section 5.4 and illustrated by the differing plots in Figure 5.1.

We now show how to modify the bisection and generalized eigenvalue methods to exploit the fixed block.

5.1. Bisection method. Let us look more closely at the Cholesky-based test for definiteness in the case where $S(\alpha)$ is given by (5.4). If $R = \left[\begin{smallmatrix} R_{11} & R_{12} \\ 0 & R_{22} \end{smallmatrix}\right]$ is the Cholesky factor of $S(\alpha)$ then

1. R_{11} is the Cholesky factor of the fixed block A.
2. R_{12} is the solution of the multiple right-hand side triangular system $R_{11}^T R_{12} = (1-\alpha)Y$.
3. R_{22} is the Cholesky factor of $\alpha I + (1-\alpha)B - R_{12}^T R_{12}$.

Note that R_{11} is independent of α and so needs to be computed only once. Also, since $R_{12} = (1-\alpha)R_{11}^{-T}Y$, once we have computed the solution X of $R_{11}^T X = Y$ then for each α we do not need to solve a linear system for R_{12}, but can instead set $R_{12} = (1-\alpha)X$. Hence, to determine if the matrix $S(\alpha)$ is positive definite or not for a given α we attempt to compute the Cholesky factor of the matrix $\alpha I + (1-\alpha)B - (1-\alpha)^2 X^T X$ (which of course is the Schur complement of A in $S(\alpha)$).

Taking all this into account, our optimized bisection algorithm for the case when A is positive definite is as follows.

ALGORITHM 5.1 (Bisection method). *Given the indefinite matrix M_0 in (5.1) with positive definite $(1,1)$ block A and a convergence tolerance* tol*, this algorithm uses the bisection method with Cholesky factorization to compute the optimal shrinking parameter α_* defined by (5.3) for the target matrix (5.2).*

```
$\alpha_\ell = 0, \;\; \alpha_r = 1$
Compute $R_{11}$, the Cholesky factor of $A$.
Compute the solution $X$ of $R_{11}^T X = Y$ and form $Z = X^T X$.
```

```
while $\alpha_r - \alpha_\ell > \mathrm{tol}$
        $\alpha_m = (\alpha_\ell + \alpha_r)/2$
        $T = \alpha_m I + (1-\alpha_m)B - (1-\alpha_m)^2 Z$
        if the Cholesky factorization of $T$ breaks down
            $\alpha_\ell = \alpha_m$
        else
            $\alpha_r = \alpha_m$
        end
end
$\alpha_* = \alpha_r$
```

The cost of Algorithm 5.1 is at most $m^3/3 + m^2n + n^2m + n^3\lceil|\log_2 \mathrm{tol}|\rceil/3$ flops.

5.2. Generalized eigenvalue problem. Recall from section 3.3 that we are looking for the smallest generalized eigenvalue of the definite pencil $M_0 - \mu M_1$, with M_0 and M_1 now given by (5.1) and (5.2). If $A = R_{11}^T R_{11}$ is the Cholesky factorization then

$$M_0 - \mu M_1 = \begin{bmatrix} A - \mu A & Y \\ Y^T & B - \mu I \end{bmatrix} = \begin{bmatrix} R_{11}^T & 0 \\ 0 & I \end{bmatrix} \left(\begin{bmatrix} I & R_{11}^{-T} Y \\ Y^T R_{11}^{-1} & B \end{bmatrix} - \mu I \right) \begin{bmatrix} R_{11} & 0 \\ 0 & I \end{bmatrix}.$$

Hence we obtain a standard symmetric eigenvalue problem for the matrix

$$C = \begin{bmatrix} I & R_{11}^{-T} Y \\ Y^T R_{11}^{-1} & B \end{bmatrix}, \tag{5.5}$$

at the cost of one Cholesky factorization and one multiple right-hand side triangular system solve.

In summary, we have the following algorithm.

Algorithm 5.2 (Generalized eigenvalue method). *Given the indefinite matrix M_0 in (5.1) with positive definite $(1,1)$ block A this algorithm uses the generalized eigenvalue interpretation to compute the optimal shrinking parameter α_* defined by (5.3) for the target matrix (5.2).*

```
Compute $R_{11}$, the Cholesky factor of $A$.
Compute the solution $X$ of $R_{11}^T X = Y$ and form $C$ from (5.5).
Find $\mu_*$, the smallest eigenvalue of the matrix $C$, by tridiagonalization
    (exploiting the identity block) followed by the bisection method.
$\alpha_* = \mu_*/(\mu_* - 1)$
```

The cost of Algorithm 5.2 is at most $m^3/3 + m^2n + 4(m+n)^3/3$ flops.

The essential difference between Algorithm 3.4 and Algorithm 5.2 is that we need the Cholesky factorization of M_1 in the former but only that of A in the latter.

5.3. A parametrized problem. In applications it may be required that a correlation matrix is strictly positive definite. When the $(1,1)$ block A is positive definite we therefore generalize the problem to

$$\alpha_*^\psi = \min\{\, \alpha \in [0,1] \colon f(\alpha) \ge \psi := \theta \lambda_{\min}(A) \,\}, \tag{5.6}$$

where θ is a parameter. For $\theta = 0$ we have the original problem. To obtain an upper bound on the possible choices of θ we next show that the function f in the fixed block case attains its maximum value at $\alpha = 1$. Note that, since $S(1) = \mathrm{diag}(A, I)$ we have

$$f(1) = \lambda_{\min}\left(S(1)\right) = \min\{\lambda_{\min}(A), 1\} = \lambda_{\min}(A), \tag{5.7}$$

because $\lambda_{\min}(A) \le 1$ by (2.4) for any symmetric matrix with unit diagonal.

LEMMA 5.3. *For $S(\alpha)$ in (5.4) with A positive definite the function f defined by (2.3) is nondecreasing on* $[0,1]$.

Proof. Since $f(0) < 0$, $f(1) = \lambda_{\min}(A) > 0$ and f is concave and continuous, it is sufficient to show that for every $\alpha \in [0,1]$ we have $f(\alpha) \le f(1)$, that is, $\max_{\alpha\in[0,1]} f(\alpha) = f(1)$.

From (5.4), since A is a leading principal submatrix of $S(\alpha)$, for every α we have, using (2.4), $\lambda_{\min}(A) \ge \lambda_{\min}(S(\alpha)) = f(\alpha)$. Since $f(1) = \lambda_{\min}(A)$ by (5.7), we have $f(\alpha) \le f(1)$. □

From the proof we have $f(\alpha) \le \lambda_{\min}(A)$. Therefore θ in (5.6) should be restricted to $[0,1]$ and hence $\psi \in [0,1]$.

Clearly, finding α_*^{ψ} is equivalent to finding the minimal α such that the matrix

$$S_\psi(\alpha) = S(\alpha) - \psi I \tag{5.8}$$

is positive semidefinite, and since $f_\psi(\alpha) = \lambda_{\min}(S_\psi(\alpha)) = f(\alpha) - \psi$ it follows that f_ψ has all the same properties as f: it is a concave and nondecreasing function on $[0,1]$.

With $A_\psi = A - \psi I$ and $B_\psi = B - \psi I$, we can write $S_\psi(\alpha)$ from (5.8) as

$$S_\psi(\alpha) = \begin{bmatrix} A_\psi & (1-\alpha)Y \\ (1-\alpha)Y^T & \alpha(1-\psi)I + (1-\alpha)B_\psi \end{bmatrix}.$$

For $\psi < 1$ the matrix A_ψ is positive definite and with only minor changes the methods from sections 5.1 and 5.2 for computing α_* in (5.3) can be used to compute α_*^{ψ} in (5.6).

The extreme case, when $\theta = 1$, significantly changes the nature of the problem. Here we are asking that α_* is such that $\lambda_{\min}(S(\alpha_*)) = \lambda_{\min}(A)$ and it follows that the matrix A_ψ is singular and positive semidefinite. In this case, $\alpha_* = 1$ might be the only solution or all values on the interval $[\alpha_*, 1]$, with $\alpha_* < 1$, might be solutions. These two cases are illustrated in plots (c) and (d) of Figure 5.1, with A there representing A_ψ.

In the next section we discuss the problems, both theoretical and computational, that arise from the singularity of the leading block in $S(\alpha)$.

5.4. Singular A. We now suppose that A in (5.1) is positive semidefinite and singular. For the bisection method, none of the computational savings discussed in section 5.1 are now applicable, since they were derived under the assumption that A is positive definite. We still have the basic Algorithm 3.1, but f may now be zero on an interval $[\alpha_*, 1]$ (see Figure 5.1(d)). In this case we cannot use the standard Cholesky factorization, since its success or failure for a singular positive semidefinite matrix is unpredictable. Instead we need to use the Cholesky factorization with complete pivoting. If the factorization runs to completion with positive pivots we declare the matrix positive semidefinite. If the factorization terminates with a nonpositive pivot, we declare the matrix positive semidefinite if the norm of the remaining block S_k (the Schur complement) satisfies $\|S_k\| \le c_n u \|A\|$, where c_n is a constant and u is the unit roundoff, and indefinite otherwise. A weakness of this approach is that c_n could potentially have to be of order 4^{n-1} in order to not misdiagnose a positive semidefinite matrix as indefinite [16], [17, sec. 10.3.2].

The case of an interval of zeros is not problematic for the Newton method. However, the method might run into difficulties when $\alpha = 1$ is the only solution and the function f is slowly increasing near that point, because then the computed iterates might leave the [0,1] bracket. Therefore the Newton method should be safeguarded.

The most severe difficulties arise in the generalized eigenvalue method. If A is singular then M_1 is singular and we no longer have a definite pencil; moreover, if f has infinitely many zeros then the pencil (3.3) is singular, which means that every α is a generalized eigenvalue and we cannot characterize α_* as before.

Our preferred way to handle the case of singular A is to employ a deflation method that reduces the problem to the nonsingular case. As a bonus, this analysis also allows us to distinguish the case when f has infinitely many zeros in $[0,1]$ from the case when its only zero is 1.

Since A is singular and positive semidefinite it has the eigenvalue decomposition $A = QDQ^T$, where Q is orthogonal and $D = \operatorname{diag}(0, D_+)$, with D_+ a nonsingular diagonal matrix of size $r = \operatorname{rank}(A)$, containing all the positive eigenvalues of A. Then

$$\begin{aligned} S(\alpha) &= \alpha \begin{bmatrix} A & 0 \\ 0 & I \end{bmatrix} + (1-\alpha) \begin{bmatrix} A & Y \\ Y^T & B \end{bmatrix} \\ &= \begin{bmatrix} Q & 0 \\ 0 & I \end{bmatrix} \left(\alpha \begin{bmatrix} D & 0 \\ 0 & I \end{bmatrix} + (1-\alpha) \begin{bmatrix} D & Q^TY \\ Y^TQ & B \end{bmatrix} \right) \begin{bmatrix} Q & 0 \\ 0 & I \end{bmatrix}^T \\ &= \operatorname{diag}(Q,I)\widetilde{S}(\alpha) \operatorname{diag}(Q,I)^T, \end{aligned}$$

with

$$\begin{aligned} \widetilde{S}(\alpha) &= \alpha \left[\begin{array}{cc|c} 0 & 0 & \\ 0 & D_+ & 0 \\ \hline \multicolumn{2}{c|}{0} & I \end{array} \right] + (1-\alpha) \left[\begin{array}{cc|c} 0 & 0 & \\ 0 & D_+ & Q^TY \\ \hline \multicolumn{2}{c|}{Y^TQ} & B \end{array} \right] \\ &= \left[\begin{array}{cc|c} 0 & 0 & \\ 0 & D_+ & (1-\alpha)Q^TY \\ \hline \multicolumn{2}{c|}{(1-\alpha)Y^TQ} & \alpha I + (1-\alpha)B \end{array} \right]. \end{aligned}$$

A necessary condition for $\widetilde{S}(\alpha)$ to be positive semidefinite is that the first $m-r$ rows of $(1-\alpha)Q^TY$ are zero. If the first $m-r$ rows of Q^TY are not zero then $\alpha_* = 1$. Otherwise, $\alpha_* < 1$ and α_* is the smallest α such that

$$\widetilde{S}_+(\alpha) = \begin{bmatrix} D_+ & (1-\alpha)Z \\ (1-\alpha)Z^T & \alpha I + (1-\alpha)B \end{bmatrix}$$

is positive semidefinite, where Z comprises the last r rows of Q^TY. Since the leading (1,1) block of this matrix is now positive definite we can find α_* by either of the methods from sections 5.1 and 5.2 with $S(\alpha)$ replaced by $\widetilde{S}_+(\alpha)$.

Note that the condition that the first $m-r$ rows of Q^TY are zero means that each column of Y is in the column space of A.

5.5. Generalization to multiple fixed blocks. The problem of this section generalizes naturally to applications where a large correlation matrix needs to be constructed from blocks, as in the risk aggregation problem described in section 1. Shrinking can easily be used to solve this problem by choosing as target the matrix comprising the diagonal blocks of the large matrix: $M_1 = \operatorname{diag}(A_{11}, A_{22}, \dots, A_{kk})$. As in the case of keeping just one block fixed, α_* is characterized as the minimal elementwise relative change in the cross-correlations.

When $k = 2$ it is easy to show that the optimized bisection algorithm is a simple modification of Algorithm 5.1, where in step 6 the matrix T is now $T = A_{22} - (1 -$

$\alpha_m)^2 Z$. For optimal efficiency the matrix should be reordered, if necessary, so that the larger of the two diagonal blocks is in the $(1,1)$ position. For the generalized eigenvalue method, Algorithm 3.4 leads to computing the smallest eigenvalue of

$$C = \begin{bmatrix} I & Z \\ Z^T & I \end{bmatrix},$$

where $Z = R_{11}^{-T} Y R_{22}^{-1}$ is formed by solving linear systems with the Cholesky factors R_{11} of A_{11} and R_{22} of A_{22}. Note that the required smallest eigenvalue of C is equal to $1-\sigma_*$, where σ_* is the largest singular value of the matrix Z, so instead of computing the smallest eigenvalue of a matrix of order $m+n$ we can compute the largest singular value of an $m \times n$ matrix. For $k > 2$, the general algorithms from section 3 should be used.

When some of the diagonal blocks are singular, deflation analogous to that in section 5.4 can be done by employing the eigenvalue decompositions of each singular diagonal block.

6. Numerical experiments.

Our tests were carried out in MATLAB R2014a on a machine with an Intel Core i7-4910MQ 2.90GHz processor and 16GB RAM.

We first compare the performance of our methods on a correlation matrix problem with a fixed block. We generate M_0 in (5.1) and M_1 in (5.2) by forming $A \in \mathbb{R}^{m\times m}$ using the MATLAB function `gallery('randcorr',m)` and the elements of the blocks $Y \in \mathbb{R}^{m\times n}$ and $B \in \mathbb{R}^{n\times n}$ are taken from the uniform distribution on $[-1,1]$, with B symmetric and forced to have unit diagonal. The size $N = m+n$ of M_0 varies from 300 to 1500 and the test matrices are split into three groups. In the first group the matrices A and B (the diagonal blocks of M_0) are of the same size, in the second A is twice the size of B, and in the third B is twice the size of A. Unless specified otherwise, we use tolerance tol $= 10^{-6}$, which is small enough for most practical applications.

We use the following algorithms.

1. `bisection`: Algorithm 3.1 with Cholesky factorization.
2. `bisection_fb`: Algorithm 5.1, the optimized bisection algorithm for the fixed block case.
3. `newton`: Algorithm 3.3. On each iteration the required eigenvector is computed by tridiagonalization followed by the bisection method (with the same tolerance as for the Newton iteration itself), using routines from the NAG Toolbox for MATLAB Mark 24 [27].
4. `GEP`: Algorithm 3.4, the algorithm based on solving a generalized eigenvalue problem. The tridiagonalization and bisection is again done using the NAG Toolbox.
5. `GEP_fb`: Algorithm 5.2, the optimized generalized eigenvalue problem algorithm for the fixed block case.

The computation times for the five methods averaged over 10 matrices of each size are presented in Table 6.1. The average number of steps for `newton` varies from 7 to 10 and the bisection methods always take 20 steps.

The experiments confirm the merit of using the optimized versions of bisection and the generalized eigenvalue method in applications where we keep a block fixed. Newton's method is the slowest of the three methods. `GEP` is a little faster than `bisection`, while `bisection_fb` is faster than `GEP_fb` for $m = n$ and $m = 2n$ and of similar speed for $n = 2m$.

Table 6.1

Computation times in seconds for the three general algorithms and the two algorithms optimized for the fixed block problem, for invalid correlation matrices of size $m + n$ with fixed leading block of size m.

(m, n)	`bisection`	`bisection_fb`	`GEP`	`GEP_fb`	`newton`
(150,150)	0.0069	0.0028	0.0039	0.0028	0.0194
(300,300)	0.0384	0.0091	0.0224	0.0147	0.1052
(450,450)	0.1029	0.0206	0.0642	0.0399	0.3055
(600,600)	0.2143	0.0435	0.1474	0.0895	0.6242
(750,750)	0.3835	0.0815	0.2913	0.1819	1.4204
(200,100)	0.0075	0.0017	0.0039	0.0031	0.0189
(400,200)	0.0405	0.0058	0.0230	0.0170	0.1215
(600,300)	0.1087	0.0121	0.0679	0.0472	0.3053
(800,400)	0.2381	0.0227	0.1571	0.1093	0.7699
(1000,500)	0.3848	0.0382	0.2744	0.1911	1.5115
(100,200)	0.0067	0.0043	0.0035	0.0025	0.0166
(200,400)	0.0308	0.0138	0.0180	0.0114	0.0827
(300,600)	0.0908	0.0367	0.0528	0.0313	0.2531
(400,800)	0.2068	0.0797	0.1304	0.0727	0.5445
(500,1000)	0.3306	0.1426	0.2667	0.1588	1.1879

To illustrate the effect of weighting, we consider an example with M_0 and W defined by

$$M_0 = \begin{bmatrix} 1.000 & 0.900 & 0.450 & 0.300 & 0.225 \\ 0.900 & 1.000 & 0.900 & 0.450 & 0.300 \\ 0.450 & 0.900 & 1.000 & 0.900 & 0.450 \\ 0.300 & 0.450 & 0.900 & 1.000 & 0.900 \\ 0.225 & 0.300 & 0.450 & 0.900 & 1.000 \end{bmatrix}, \quad W = \begin{bmatrix} 1 & 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0.5 \\ 0 & 0 & 1 & 0.5 & 1 \end{bmatrix}.$$

The eigenvalues of M_0 are, to the digits shown, $-0.18, 0.05, 0.50, 1.27, 3.36$, therefore M_0 is indefinite. We see that

$$M_1 = W \circ M_0 = \begin{bmatrix} 1.00 & 0.90 & 0.00 & 0.00 & 0.00 \\ 0.90 & 1.00 & 0.00 & 0.00 & 0.00 \\ 0.00 & 0.00 & 1.00 & 0.00 & 0.45 \\ 0.00 & 0.00 & 0.00 & 1.00 & 0.45 \\ 0.00 & 0.00 & 0.45 & 0.45 & 1.00 \end{bmatrix},$$

with eigenvalues $0.10, 0.36, 1.00, 1.64, 1.90$. Hence, M_1 is a valid target. Using `bisection` we obtain $\alpha_* = 0.24$ and

$$S(\alpha_*) = \begin{bmatrix} \mathbf{1.000} & \mathbf{0.900} & 0.343 & 0.228 & 0.171 \\ \mathbf{0.900} & \mathbf{1.000} & 0.685 & 0.343 & 0.228 \\ 0.343 & 0.685 & \mathbf{1.000} & 0.685 & \mathbf{0.450} \\ 0.228 & 0.343 & 0.685 & \mathbf{1.000} & 0.793 \\ 0.171 & 0.228 & \mathbf{0.450} & 0.793 & \mathbf{1.000} \end{bmatrix},$$

with eigenvalues $0.00, 0.16, 0.52, 1.37, 2.95$. Note that the elements corresponding to weight $w_{ij} = 1$ (typeset in bold) are unchanged, as required by the interpretation of weights.

Table 6.2

Times in seconds to compute shrinking solution by `bisection` *and* `GEP` *and nearest correlation matrix using* `g02aa`*, for tolerances* 10^{-3} *and* 10^{-6}*, and Frobenius norm distances to original matrix.*

	Time						Distance	
n	`bisection`		`GEP`		`g02aa`		shrinking	NCM
	1e-3	1e-6	1e-3	1e-6	1e-3	1e-6		
1399	0.17	0.28	0.14	0.13	3.66	4.37	321.03	21.03
3120	1.01	2.22	2.44	2.46	28.08	34.31	178.71	5.44
2798	0.70	1.61	1.69	1.69	44.24	50.88	1221.21	1089.51
4519	2.29	4.99	8.10	8.00	220.88	234.68	1761.50	1631.52
6240	7.13	17.50	21.64	21.84	447.32	449.91	2578.10	2446.80

To explore this example further, denote the elements of M_0 and $S(\alpha_*)$ by s_{ij} and s'_{ij}, respectively. Then $s_{34} = 0.9 = s_{45}$, but since $w_{34} = 0$ and $w_{45} = 0.5$ the elements s'_{34} and s'_{45} are not the same. The relative change in the $(3,4)$ element is

$$\frac{s_{34} - s'_{34}}{s_{34}} = 0.24 = \alpha_* = \alpha_*(1 - w_{34}),$$

but that in the $(4,5)$ element is

$$\frac{s_{45} - s'_{45}}{s_{45}} = 0.12 = \frac{\alpha_*}{2} = \alpha_*(1 - w_{45}),$$

confirming that each element s'_{ij} in $S(\alpha_*)$ was obtained by multiplying the corresponding element s_{ij} in M_0 by $\alpha_*(1 - w_{ij})$.

In our next example, we use two invalid correlation matrices from the finance industry. The matrix C_1 is of order 1399 and the matrix C_2 is of order 3120. We use the identity matrix as the target in the shrinking method, thus we are not fixing any off-diagonal elements. We compare the execution times of `bisection` and `GEP` with that for computation of the nearest correlation matrix (NCM) to C_1 and C_2 by NAG code `g02aa/nag_nearest_correlation` which implements a Newton method [4], [30]. We also use three further matrices constructed as block 2×2 matrices with diagonal blocks C_1 and C_1, C_1 and C_2, and C_2 and C_2, with remaining off-diagonal elements from the random uniform distribution on $[-1, 1]$. Convergence tolerances of both 10^{-3} and 10^{-6} are taken for `bisection` and `g02aa`, as well as for the bisection part of `GEP`. The times are shown in Table 6.2. The shrinking solution is computed one to two orders of magnitude faster than the NCM. It is clear that `g02aa` and `GEP` do not benefit significantly from a relaxed tolerance, whereas the time for `bisection` is proportional to the logarithm of the tolerance. The table also shows that the Frobenius norm distances from the original matrix to the shrinking solution range from being similar to the distance to the NCM to much larger than it.

Our final experiment provides some insight into how the Frobenius norm distance from the original matrix to the shrinking solution compares with the distance to the NCM when the smallest eigenvalue varies in size. We take for M_0 a random symmetric indefinite matrix with unit diagonal of size 500, constructed by a diagonal scaling of a random orthogonal similarity applied to a diagonal matrix D; the diagonal elements of D are generated from the uniform distribution on $[0, 1]$ and half of them are multiplied by -10^{-p} for some p. We generate 10 random target correlation matrices M_1 and apply `bisection`. For each M_0, Table 6.3 shows the average shrinking parameter, the

Table 6.3

Comparison of the distances in the Frobenius norm of the NCM and the solution computed by shrinking for matrices M_0 of size 500 with varying order of magnitude for the smallest eigenvalue.

			Distance	
$\lambda_{\min}(M_0)$	Avg. α_*	NCM	shrinking (max)	shrinking (I)
-4.6635e-1	6.0586e-1	5.5041e0	2.3506e1	1.0936e1
-4.2839e-2	9.7077e-2	5.0532e-1	3.5461e0	1.2367e0
-4.2466e-3	1.0144e-2	4.9173e-2	3.5748e-1	1.2255e-1
-4.0378e-4	9.4919e-4	4.6492e-3	3.3126e-2	1.1789e-2
-4.3278e-5	1.0900e-4	5.3498e-4	3.9736e-3	1.2897e-3
-4.0634e-6	1.0014e-5	4.5978e-5	3.4107e-4	1.3969e-4

NCM distance, the maximum distance for the shrinking solution, and the distance for shrinking with $M_1 = I$. We see that the distance with $M_1 = I$ is smaller than the worst-case for the random targets M_1 and the shrinking distance is one order of magnitude larger than the NCM distance. This experiment gives some feel for the trade-off between the speed of shrinking versus the optimality of the NCM as measured by distance, at least for the case where there are no fixed elements.

It is clear from the last two experiments that in applications where it might not be necessary to compute the NCM, or where an elementwise approach might be preferable, shrinking presents an attractive and much faster alternative.

7. Concluding remarks. The motivation for this work was the growing number of applications in which matrices that are supposed to be (semi)definite turn out to be indefinite. We aimed to develop an alternative to the popular, but relatively expensive, approach of replacing the given matrix by the nearest positive semidefinite matrix or nearest correlation matrix. We have shown that shrinking is an attractive way of restoring definiteness. It can be achieved in at least three different ways, all of which are straightforward to implement. Weights can be incorporated in the target matrix without any effect on the solution techniques. Shrinking can also take advantage of fixed diagonal blocks, as occur in stress testing, for example.

Of our three shrinking methods we favor the bisection and generalized eigenvalue methods. Bisection is perhaps preferable for convergence tolerances of 10^{-6} and larger, whereas the generalized eigenvalue method is preferable for more stringent tolerances, since its cost is essentially independent of the tolerance. Bisection also has the advantage that it produces a numerically semidefinite matrix (one for which Cholesky factorization succeeds).

The problems for which we believe shrinking is of interest range in size from order 10–100, as arise for example in foreign exchange trading, and which may need to be solved thousands of times in a simulation, to orders in the thousands or millions, as for example in bioinformatics [35]. In the case of invalid correlation matrices an attractive feature of shrinking is that it is at least an order of magnitude faster than computing the nearest correlation matrix.

An implementation of Algorithm 3.1 is to appear in Mark 25 of the NAG Library (2015) as routine `g02anf` [28]. MATLAB implementations of the algorithms discussed here are available at `https://github.com/higham/shrinking` and Python implementations, which do not require access to the NAG Library, are available at `https://github.com/vsego/shrinking`.

REFERENCES

[1] Greg Anderson, Lisa Goldberg, Alec N. Kercheval, Guy Miller, and Kathy Sorge. On the aggregation of local risk models for global risk management. *Journal of Risk*, 8(1):25–40, 2005.

[2] Vineer Bhansali and Mark B. Wise. Forecasting portfolio risk in normal and stressed markets. *J. Risk*, 4(1):91–106, 2001.

[3] Rüdiger Borsdorf. A Newton algorithm for the nearest correlation matrix. M.Sc. Thesis, The University of Manchester, Manchester, UK, September 2007. 151 pp. MIMS EPrint 2008.49, Manchester Institute for Mathematical Sciences, The University of Manchester.

[4] Rüdiger Borsdorf and Nicholas J. Higham. A preconditioned Newton algorithm for the nearest correlation matrix. *IMA J. Numer. Anal.*, 30(1):94–107, 2010.

[5] Charles R. Crawford and Yiu Sang Moon. Finding a positive definite linear combination of two Hermitian matrices. *Linear Algebra Appl.*, 51:37–48, 1983.

[6] Philip I. Davies, Nicholas J. Higham, and Françoise Tisseur. Analysis of the Cholesky method with iterative refinement for solving the symmetric definite generalized eigenproblem. *SIAM J. Matrix Anal. Appl.*, 23(2):472–493, 2001.

[7] Hakan Demirtas, Donald Hedeker, and Robin J. Mermelstein. Simulation of massive public health data by power polynomials. *Statistics in Medicine*, 31(27):3337–3346, 2012.

[8] James W. Demmel. *Applied Numerical Linear Algebra*. Society for Industrial and Applied Mathematics, Philadelphia, PA, USA, 1997. xi+419 pp. ISBN 0-89871-389-7.

[9] Susan J. Devlin, R. Gnanadesikan, and J. R. Kettenring. Robust estimation and outlier detection with correlation coefficients. *Biometrika*, 62(3):531–545, 1975.

[10] Guzmán Díaz, Javier Gómez-Aleixandre, and José Coto. Statistical characterization of aggregated wind power from small clusters of generators. *International Journal of Electrical Power & Energy Systems*, 62:273 – 283, 2014.

[11] Christopher C. Finger. A methodology to stress correlations. *RiskMetrics Monitor*, Fourth Quarter:3–11, 1997.

[12] Scott Gerlt, Wyatt Thompson, and Douglas J. Miller. Exploiting the relationship between farm-level yields and county-level yields for applied analysis. *Journal of Agricultural and Resource Economics*, 39(2):253–270, 2014.

[13] Gene H. Golub and Charles F. Van Loan. *Matrix Computations*. Fourth edition, Johns Hopkins University Press, Baltimore, MD, USA, 2013. xxi+756 pp. ISBN 978-1-4214-0794-4.

[14] Chun-Hua Guo, Nicholas J. Higham, and Françoise Tisseur. An improved arc algorithm for detecting definite Hermitian pairs. *SIAM J. Matrix Anal. Appl.*, 31(3):1131–1151, 2009.

[15] Nicholas J. Higham. Computing a nearest symmetric positive semidefinite matrix. *Linear Algebra Appl.*, 103:103–118, 1988.

[16] Nicholas J. Higham. Analysis of the Cholesky decomposition of a semi-definite matrix. In *Reliable Numerical Computation*, M. G. Cox and S. J. Hammarling, editors, Oxford University Press, 1990, pages 161–185.

[17] Nicholas J. Higham. *Accuracy and Stability of Numerical Algorithms*. Second edition, Society for Industrial and Applied Mathematics, Philadelphia, PA, USA, 2002. xxx+680 pp. ISBN 0-89871-521-0.

[18] Nicholas J. Higham. Computing the nearest correlation matrix—A problem from finance. *IMA J. Numer. Anal.*, 22(3):329–343, 2002.

[19] Alec N. Kercheval. Optimal covariances in risk model aggregation. In *Proceedings of the Third IASTED International Conference on Financial Engineering and Applications*, Calgary, 2006, pages 30–35. ACTA Press.

[20] Paul H. Kupiec. Stress testing in a value at risk framework. *J. Derivatives*, 6(1):7–24, 1998. Reprinted as [21].

[21] Paul H. Kupiec. Stress testing in a value at risk framework. In *Risk Management: Value at Risk and Beyond*, M. A. H. Dempster, editor, Cambridge University Press, 2002, chapter 3, pages 76–99.

[22] Olivier Ledoit and Michael Wolf. Improved estimation of the covariance matrix of stock returns with an application to portfolio selection. *J. Empirical Finance*, 10(5):603–621, 2003.

[23] Olivier Ledoit and Michael Wolf. Honey, I shrunk the sample covariance matrix. *J. Portfolio Management*, 30(4):110–119, 2004.

[24] Olivier Ledoit and Michael Wolf. A well-conditioned estimator for large-dimensional covariance matrices. *J. Multivariate Analysis*, 88(2):365–411, 2004.

[25] Craig Lucas. Computing nearest covariance and correlation matrices. M.Sc. Thesis, University of Manchester, Manchester, England, October 2001. 68 pp.

[26] Hossein Mohammadi, Abbas Seifi, and Toomaj Foroud. A robust Kriging model for predicting

accumulative outflow from a mature reservoir considering a new horizontal well. *Journal of Petroleum Science and Engineering*, 82-83:113–119, 2012.

[27] NAG Toolbox for MATLAB. NAG Ltd., Oxford, UK. `http://www.nag.co.uk`.

[28] NAG Library. NAG Ltd., Oxford, UK. `http://www.nag.co.uk`.

[29] Kawee Numpacharoen. Weighted average correlation matrices method for correlation stress testing and sensitivity analysis. *Journal of Derivatives*, 21(2):67–74, 2013.

[30] Hou-Duo Qi and Defeng Sun. A quadratically convergent Newton method for computing the nearest correlation matrix. *SIAM J. Matrix Anal. Appl.*, 28(2):360–385, 2006.

[31] Hou-Duo Qi and Defeng Sun. An augmented Lagrangian dual approach for the H-weighted nearest correlation matrix problem. *IMA J. Numer. Anal.*, 31:491–511, 2011.

[32] HouDuo Qi and Defeng Sun. Correlation stress testing for value-at-risk: an unconstrained convex optimization approach. *Comput Optim Appl*, 45(2):427–462, 2010.

[33] Richard D. Ray and Bruce C. Douglas. Experiments in reconstructing twentieth-century sea levels. *Progress in Oceanography*, 91:496–515, 2011.

[34] Riccardo Rebonato and Peter Jäckel. The most general methodology for creating a valid correlation matrix for risk management and option pricing purposes. *Journal of Risk*, 2 (2):17–27, 2000.

[35] Juliane Schäfer and Korbinian Strimmer. A shrinkage approach to large-scale covariance matrix estimation and implications for functional genomics. *Statistical Applications in Genetics and Molecular Biology*, 4(1), 2005. Article 32.

[36] I. D. Stewart, I. M. S. White, A. R. Gilmour, R. Thompson, J. A. Woolliams, and S. Brotherstone. Estimating variance components and predicting breeding values for eventing disciplines and grades in sport horses. *Animal*, 6(9):1377–1388, 2012.

[37] Giulia Vilone, Damien Comiskey, Fanny Heraud, and Cian O'Mahony. Statistical method to assess usual dietary intakes in the European population. *Food Additives & Contaminants: Part A*, 31(10):1639–1651, 2014.