# A SEQUENTIAL SEMISMOOTH NEWTON METHOD FOR THE NEAREST LOW-RANK CORRELATION MATRIX PROBLEM *

**Article** · May 2020

CITATIONS
0

READS
56

**2 authors:**

Qing-Na Li
Beijing Institute of Technology
**37** PUBLICATIONS **94** CITATIONS
SEE PROFILE

Houduo Qi
University of Southampton
**86** PUBLICATIONS **1,861** CITATIONS
SEE PROFILE

SIAM J. OPTIM.
Vol. 21, No. 4, pp. 1641–1666


# A SEQUENTIAL SEMISMOOTH NEWTON METHOD FOR THE NEAREST LOW-RANK CORRELATION MATRIX PROBLEM*

QINGNA LI† AND HOU-DUO QI‡

**Abstract.** Based on the well-known result that the sum of the largest eigenvalues of a symmetric matrix can be represented as a semidefinite programming problem (SDP), we formulate the nearest low-rank correlation matrix problem as a nonconvex SDP and propose a numerical method that solves a sequence of least-square problems. Each of the least-square problems can be solved by a specifically designed semismooth Newton method, which is shown to be quadratically convergent. The sequential method is guaranteed to produce a stationary point of the nonconvex SDP. Our numerical results demonstrate the high efficiency of the proposed method on large scale problems.

**Key words.** low-rank correlation matrix, quadratic semidefinite programming, semismooth Newton method, quadratic convergence, constraint nondegeneracy

**AMS subject classifications.** 49M45, 90C25, 90C33

**DOI.** 10.1137/090771181

## 1. Introduction.

In this paper, we introduce a fast algorithm for the problem of computing the nearest correlation matrix under an arbitrary rank constraint:

$$\begin{array}{ll} \min\limits_{X\in\mathcal{S}^n} & \frac{1}{2}\|X-C\|^2 \\ \text{s.t.} & \operatorname{diag}(X)=e,\ X\succeq 0, \\ & \operatorname{rank}(X)\le r, \end{array} \tag{1}$$

where $C\in\mathcal{S}^n$ is given; $\mathcal{S}^n$ is the space of $n\times n$ symmetric matrices endowed with the standard trace inner product; $\|\cdot\|$ is the induced norm (i.e., the Frobenius norm); $\mathcal{S}^n_+$ denotes the cone of all positive semidefinite matrices in $\mathcal{S}^n$; $X\succeq 0$ means $X\in\mathcal{S}^n_+$; $\operatorname{diag}(X)$ is the vector formed by the diagonal elements of $X$; $e$ is the vector of all ones in $\mathbb{R}^n$; $\operatorname{rank}(X)$ is the rank of $X$; and $r\le n$ is a given integer.

The first constraint in (1) defines the set of all correlation matrices in $\mathcal{S}^n_+$, and the second constraint stipulates that only those correlation matrices that have rank less than $r$ are feasible. On top of those conditions, we seek the nearest candidate to a given matrix $C$. Unless $r=n$ (so that the last constraint becomes superfluous), problem (1) is a nonconvex problem.

Problem (1) has important applications in finance, where the input matrix $C$ is often a known correlation matrix but with rank larger than $r$. Interested readers may refer to Brigo and Mercurio [6, sect. 6.9], Rebonato [31, sect. 9] and [32], Grubišić and Pietersz [12], Pietersz and Groenen [25], Wu [38], and Zhang and Wu [39] (to just name a few) for concrete examples and a few proposed numerical algorithms, some of which will be briefly discussed shortly. In fact, the literature review section [12, sect. 2] refers to 17 papers and books addressing the problem.

*Received by the editors September 16, 2009; accepted for publication (in revised form) September 23, 2011; published electronically December 15, 2011.
http://www.siam.org/journals/siopt/21-4/77118.html

†College of Mathematics and Econometrics, Hunan University, Changsha, Hunan, China. Current address: Institute of Computational Mathematics and Scientific/Engineering Computing, Chinese Academy of Sciences, Beijing 100190, China (liqingna@yahoo.com.cn). This author was supported by a grant from the Ministry of Education of China (309023).

‡School of Mathematics, University of Southampton, Highfield, Southampton SO17 1BJ, UK (hdqi@soton.ac.uk).

A widely adopted approach is to tackle the rank constraint $\text{rank}(X) \le r$ and the positive semidefinite constraint $X \succeq 0$ together by using the Gramian representation:

$$X = RR^T, \tag{2}$$

where $R \in \mathbb{R}^{n\times r}$. This results in (1) being a standard nonconvex programming problem with quadratic constraints (i.e., $\text{diag}(RR^T) = e$). Geometric optimization methods (including Newton and gradient types) in [12] as well as the majorization method [25] are designed to solve this nonconvex programming problem. Each of them leads to a computational package: `LRCMmin` package from the geometric approach and `Major` from the majorization approach. Both packages are in MATLAB and are publicly available. An important feature of the two packages is that there exists a sufficient condition to verify whether a local minimum (of the nonconvex programming problem) is a global minimum; see [39, Thms. 4.4 and 4.5] and [12, Thm. 6.2]. For a detailed comparison between the Lagrangian method of Zhang and Wu [39] and `Major`, see [12]. Recently, the majorization method was extended by Simon and Abell in [33] to solve problem (1) with more equality constraints.

The trigonometric parameterization method (`TPM`) takes a further step to parameterize $R$ in (2) satisfying $\text{diag}(X) = e$ by trigonometric functions through spherical coordinates. See Rebonato [30], Brigo [5], and Rapisarda, Brigo, and Mercurio [29] for more on this method. We once again refer the reader to [12, sects. 2 and 7] for a comprehensive survey on methods available for solving the low-rank problem (1). It is noted that all numerical results in the existing literature are only for problems of size up to $n = 80$. However, one may be faced with some practical problems in higher dimensions ($n \ge 500$); see [11] for such examples. Our method is capable of dealing with problems of $n$ being very large (i.e., $n \ge 1000$).

The method proposed here is strongly motivated by the semismooth Newton method of Qi and Sun [26] for the nearest correlation matrix problem [15]:

$$\begin{array}{ll} \min\limits_{X\in\mathcal{S}^n} & \frac{1}{2}\|X - C\|^2 \\ \text{s.t.} & \text{diag}(X) = e,\ X \succeq 0. \end{array} \tag{3}$$

Further development of the method can be found in Borsdorf and Higham [3], which leads to Numerical Algorithms Group (NAG) Fortran implementation. We also refer the reader to Malick [21], Boyd and Xiao [4], Toh [36], and Henrion and Malick [13] for other efficient methods for (3). In particular, [21, 4] are excellent references on the Lagrangian dual approach used in section 4.

It is obvious that numerical methods for (3) cannot be used for (1) because of the rank constraint: $\text{rank}(X) \le r$. However, the idea of the semismooth Newton method plays a very important role in our approach because each subproblem is going to be solved by it. Our approach is based on the following result, whose proof is postponed to section 2.

THEOREM 1.1. *$\overline{X} \in \mathcal{S}^n$ solves the nearest low-rank correlation matrix problem (1) if and only if there exists a matrix $\overline{U} \in \mathcal{S}^n$ such that $(\overline{X}, \overline{U})$ solves the following nonlinear semidefinite programming problem, denoted by* $\text{NSDP}_r$*:*

$$\begin{array}{lll} & \min\limits_{(X,U)\in\mathcal{S}^n\times\mathcal{S}^n} & \frac{1}{2}\|X - C\|^2 \\ & s.t. & \text{diag}(X) = e,\ X \succeq 0, \\ (\text{NSDP}_r) & & \langle X, U\rangle = n, \\ & & \langle I, U\rangle = r, \\ & & 0 \preceq U \preceq I. \end{array} \tag{4}$$

*Moreover, the optimal objective values of* (1) *and* (4) *are equal.*

NSDP$_r$ (4) is a continuous optimization reformulation of the low-rank problem (1) over symmetric matrices. Due to the bilinear term $\langle X, U\rangle = n$, it is a nonconvex problem. The main purpose of this paper is to design an efficient algorithm that finds a stationary point of NSDP$_r$ (4). We first note the complexity of (4). It has two variables $X$ and $U$, and each is of the dimension $n(n+1)/2$. Moreover, it involves three positive semidefinite cones of the same size: $X \succeq 0$, $U \succeq 0$, and $Z \succeq 0$ with $Z := I - U$. In addition to these parameters, it has the nonconvex constraint $\langle X, U\rangle = n$. It seems unwise to solve this problem directly. In fact, our initial attempt to use a semismooth Newton method based on the augmented Lagrangian to solve (4) proved unsuccessful.[1]

Our main idea to avoid dealing with those three positive semidefinite cones simultaneously is as follows. Let $X \in \mathcal{S}^n$ have the following eigenvalue decomposition:

$$X = P\mathrm{Diag}(\lambda_1, \ldots, \lambda_n)P^T, \tag{5}$$

where $PP^T = I$ and $\lambda_1 \geq \cdots \geq \lambda_n$ in nonincreasing order are eigenvalues of $X$. We solve (4) in an alternating way, i.e., by an $X$-minimization step and a $U$-minimization step. At iteration $k$, the $X$-step is to fix $U^k$ and attempt to solve the following problem, which is obtained from (4) by removing the last two constraints:

$$\begin{array}{ll} \min\limits_{X \in \mathcal{S}^n} & \frac{1}{2}\|X - C\|^2 \\ \text{s.t.} & \mathrm{diag}(X) = e,\ X \succeq 0, \\ & \langle X, U^k\rangle = n. \end{array} \tag{6}$$

We then update $X^k$ by the optimal solution of (6).

The problem of $U$-minimization is quite hard (see more comments on this in (R4), section 3). We hence propose a simple updating rule, that is, to solve the problem (given $X := X^{k+1}$)

$$\begin{array}{ll} \max\limits_{U \in \mathcal{S}^n} & \langle X, U\rangle \\ \text{s.t.} & \langle I, U\rangle = r, \\ & 0 \preceq U \preceq I, \end{array} \tag{7}$$

and a solution (see fact (iii) in section 2.2) is

$$U = P_1 P_1^T, \tag{8}$$

where $P_1$ is the submatrix of $P$ consisting of the first $r$ columns of $P$. $U^k$ is then updated by (8), where $P$ comes from the eigenvalue decomposition of $X := X^{k+1}$. The rule is right in the sense that if $(U, X)$ is a local minimum of (4), then $U$ should take the form of (8). If $X$ is not a feasible low-rank solution, the matrix $U$ given by (8) satisfies all constraints in (4) except $\langle X, U\rangle = n$. We will force this constraint to be eventually satisfied in the $X$-minimization part. As rightly pointed out by one of the referees, such $U$ is not optimal for the problem of $U$-minimization. Despite its nonoptimality of the formula, its simplicity gets paid off in speeding up the computation. This simple rule also ensures that any limiting point by the algorithm is a stationary point of (4). Our numerical results also demonstrate quite a good performance (e.g., 3 to 10 iterations of our algorithm over a range of test problems).

[1] Roughly speaking, the main reason for the failure was that the augmented Lagrangian function involves projections of three different matrices onto $\mathcal{S}^n_+$, and it took too much time to evaluate the generalized Jacobians involving the three projections.

Compared to (4), problem (6) is a standard quadratic semidefinite programming extensively studied by Toh [36], and it involves only one positive semidefinite cone $X \succeq 0$. However, a potential problem for (6) is that it may have no feasible solutions. That is, the first constraint may contradict the second constraint in (6) due to the way in which $U^k$ is defined. To show this, take

$$U^k := \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}.$$

Then it is easy to see that for any correlation matrix $X$, $\langle X, U^k \rangle = 2 < 3 = n$ , which implies that (6) is infeasible. To avoid the infeasibility issue, we propose to solve its augmented Lagrangian problem, which selectively handles the constraint $\langle X, U^k \rangle = n$ while keeping other constraints unchanged:

$$\begin{array}{ll} \min\limits_{X \in \mathcal{S}^n} & \frac{1}{2}\|X - C\|^2 - \mu\Big(\langle X, U^k \rangle - n\Big) + \frac{c}{2}\Big(\langle X, U^k \rangle - n\Big)^2 \\ \text{s.t.} & \operatorname{diag}(X) = e,\ X \succeq 0, \end{array} \tag{9}$$

where $\mu \in \mathbb{R}$ is the Lagrange multiplier corresponding to the constraint $\langle X, U^k \rangle = n$ and $c > 0$ is the penalty parameter.

The augmented Lagrangian problem is always feasible and always has a unique optimal solution because it is strongly convex in $X$. We update $X^k$ by the optimal solution of (9) and update $U^k$ accordingly. The Lagrange multiplier $\mu$ and the penalty parameter $c$ are updated according to some rules. The main task of our algorithm is to solve the subproblem (9) efficiently per iteration. We will develop a semismooth Newton method, which is motivated by the method in [26], to solve (9). Consequently, our algorithm makes use of the semismooth Newton method iteratively. We hence name it sequential semismooth Newton method.

There are two major contributions in this paper. First, the low-rank problem is reformulated as a differentiable nonlinear semidefinite programming problem (4), defined over the space of symmetric matrices. To the best of our knowledge, it was the first time that this reformulation has been studied, and it has a deep root in eigenvalue optimization. Second, we designed an algorithm that mimics the decomposition method for (4), first on the variable $X$ (i.e., $X$-minimization) and then on the variable $U$ (i.e., $U$-minimization).

The paper is organized as follows. In section 2, we characterize stationary points of the nonconvex problem (4) and include a formal proof of Theorem 1.1. We describe our algorithm in section 3 and show that every accumulation point is stationary (see Theorem 3.2). Section 4 details our semismooth Newton method. A key property that ensures the quadratic convergence (Theorem 4.3) is constraint nondegeneracy of the least-square reformulation of problem (9) (see Proposition 4.1). We conduct extensive numerical tests and comparisons with `TPM` and `Major` on a large set of problems in section 5, where we also deal with some practical issues in our implementation. The performance of the proposed method on problem (1) with more equality constraints is also included. We conclude the paper in section 6.

Notation: We use $\circ$ to denote the Hadamard product of matrices; i.e., for any $B, C \in \mathcal{S}^n$, $B \circ C = [B_{ij}C_{ij}]_{i,j=1}^n$. We let $E$ denote the matrix of all ones in $\mathcal{S}^n$. For subsets $\alpha, \beta$ of $\{1, 2, \ldots, n\}$, we denote $B_{\alpha\beta}$ as the submatrix of $B$ indexed by $\alpha$ and $\beta$, and $B_\alpha$ as the submatrix of $B$ consisting of the columns in $B$ indexed by $\alpha$. We also denote by $|\alpha|$ the cardinality of the set $\alpha$. Let $e$ denote the vector of all ones. We

sometimes use $0_n$ to denote the zero vector in $\mathbb{R}^n$. For a matrix $X \in \mathcal{S}^n$, $\mathrm{diag}(X)$ is the vector in $\mathbb{R}^n$ formed by the diagonal elements of $X$, while $\mathrm{Diag}(y)$ is the diagonal matrix formed by the vector $y \in \mathbb{R}^n$. ":=" means "define."

**2. Preliminaries.** This section includes two parts. The first is to study the characterization of stationary points of NSDP$_r$ (4). It also includes a result about *constraint nondegeneracy*, which will play an important role in the quadratic convergence analysis of the semismooth Newton method in section 4. The second part contains a formal proof of Theorem 1.1.

**2.1. Stationary points of (4).** We copy (4) below with the corresponding Lagrange multipliers to the constraints:

$$\text{(NSDP}_r\text{)} \qquad \begin{array}{lll} \min\limits_{(X,U)\in\mathcal{S}^n\times\mathcal{S}^n} & \frac{1}{2}\|X-C\|^2 & \\ \text{s.t.} & \mathrm{diag}(X) = e & (y),\\ & \langle X, U\rangle = n & (\mu),\\ & \langle I, U\rangle = r & (\nu),\\ & I - U \succeq 0 & (R),\\ & X \succeq 0,\ U \succeq 0, & \end{array} \tag{10}$$

where $y \in \mathbb{R}^n$, $\mu, \nu \in \mathbb{R}$, and $R \in \mathcal{S}^n$ are the Lagrange multipliers of the corresponding constraints. After certain elementary linear algebra, the KKT condition of NSDP$_r$ can be stated as follows:

$$\text{(KKT)} \qquad \begin{cases} \frac{1}{2}\|X-C\|^2 + \frac{1}{2}(\|X\|^2 - \|C\|^2) - e^T y - \mu n = 0,\\ \mathrm{diag}(X) = e,\\ \langle X, U\rangle = n,\\ \langle I, U\rangle = r,\\ U \succeq 0,\ -\mu X - \nu I + R \succeq 0,\ \langle U, R\rangle = \mu n + \nu r,\\ I - U \succeq 0,\ R \succeq 0,\ \langle I - U, R\rangle = 0,\\ X \succeq 0,\ X - \big(C + \mathrm{Diag}(y)\big) - \mu U \succeq 0. \end{cases} \tag{11}$$

A point $(\overline{X}, \overline{U}) \in \mathcal{S}^n \times \mathcal{S}^n$ is said to be a stationary point of NSDP$_r$ (4) if there exists Lagrange multiplier $(\bar{y}, \bar{\mu}, \bar{\nu}, \overline{R})$ such that $(\overline{X}, \overline{U}; \bar{y}, \bar{\mu}, \bar{\nu}, \overline{R})$ satisfies the KKT condition (11). We have the following result.

PROPOSITION 2.1. *Suppose* $(\overline{X}, \overline{U}; \bar{y}, \bar{\mu}, \bar{\nu}, \overline{R})$ *satisfies the KKT condition* (11) *and* $r < n$. *We must have*

$$\bar{\mu} \geq 0 \qquad \textit{and} \qquad \bar{\nu} \leq 0.$$

*Proof.* We note from (11) that

$$-\bar{\mu}\overline{X} - \bar{\nu} I + \overline{R} \succeq 0 \qquad \text{and} \qquad I - \overline{U} \succeq 0.$$

The product of the above two matrices gives

$$\begin{aligned} 0 &\leq \langle I - \overline{U}, -\bar{\mu}\overline{X} - \bar{\nu} I + \overline{R}\rangle \\ &= -\bar{\mu}(\langle I, \overline{X}\rangle - \langle \overline{U}, \overline{X}\rangle) - \bar{\nu}\langle I - \overline{U}, I\rangle + \langle I - \overline{U}, \overline{R}\rangle \\ &= -\bar{\nu}\langle I - \bar{U}, I\rangle = -\bar{\nu}(n - r). \end{aligned} \tag{12}$$

Since $r < n$, we must have from (12) that $\bar{\nu} \leq 0$. It also follows from the KKT condition (11) that

$$0 \leq \langle \overline{U}, \overline{R} \rangle = \bar{\mu} n + \bar{\nu} r,$$

which implies $\bar{\mu} \geq 0$ because we just proved $\bar{\nu} \leq 0$. □

We end this subsection with a result on constraint nondegeneracy, which is the main topic of [1] (see also [2, eq. 4.172]). The result in [37, Prop. 4.2] is proved for some special points and is extended to any correlation matrix in [27, Prop. 2.1]. To facilitate the statement, let $T_{\mathcal{S}^n_+}(X)$ be the tangent cone of $\mathcal{S}^n_+$ at $X \in \mathcal{S}^n_+$, and let $\mathrm{lin}(T_{\mathcal{S}^n_+}(X))$ be the largest linear space contained in $T_{\mathcal{S}^n_+}(X)$.

LEMMA 2.2. *Any point $X$ satisfying the correlation constraints*

$$\{\mathrm{diag}(X) = e,\ X \in \mathcal{S}^n_+\} \tag{13}$$

*is constraint nondegenerate. That is, the implication*

$$\langle \mathrm{Diag}(y),\ Z \rangle = 0 \ \ \forall\ Z \in \mathrm{lin}(T_{\mathcal{S}^n_+}(X)) \quad \Longrightarrow \quad y = 0 \tag{14}$$

*always holds at any correlation matrix $X$.*

The lemma will be used in section 4, where we will deal with a slightly more complicated set than the correlation matrix set (13). We now finish this section with a formal proof of Theorem 1.1.

**2.2. Proof of Theorem 1.1.** Theorem 1.1 is based on the following three fundamental facts:

(i) For any correlation matrix $X \in \mathcal{S}^n$,

$$\mathrm{trace}(X) = n. \tag{15}$$

(ii) For any $X \in \mathcal{S}^n_+$ admitting the eigenvalue decomposition as in (5), there holds

$$\mathrm{rank}(X) \leq r \iff \sum_{i=1}^{r} \lambda_i = \mathrm{trace}(X). \tag{16}$$

We note that $\sum_{i=1}^{r} \lambda_i$ is the sum of the first $r$ largest eigenvalues of $X$. This brings out the third fact that $\sum_{i=1}^{r} \lambda_i$ can be represented as a semidefinite programming problem (SDP) (see Overton and Womersley [23, 24] and Hiriart-Urruty and Ye [16]).

(iii) The sum of the first $r$ largest eigenvalues of $X$ can be calculated as

$$\begin{array}{lll} \displaystyle\sum_{i=1}^{r} \lambda_i = \max_{U \in \mathcal{S}^n} & \langle X, U \rangle \\ \text{s.t.} & \langle I, U \rangle = r, \\ & 0 \preceq U \preceq I, \end{array} \tag{17}$$

and a solution is given by (8).

Combination of (15), (16), and (17) yields the following proof of Theorem 1.1.

*Proof.* Let $\mathcal{F}_1$ denote the feasible region of (1) and $\mathcal{F}_2$ the $X$-component projection of the feasible region of (4):

$$\mathcal{F}_2 := \left\{ X \in \mathcal{S}^n \mid \text{there exists } U \in \mathcal{S}^n \text{ s.t. } \begin{array}{l} \mathrm{diag}(X) = e,\ X \succeq 0 \\ \langle X, U \rangle = n,\ \langle I, U \rangle = r \\ 0 \preceq U \preceq I \end{array} \right\}.$$

We prove $\mathcal{F}_1 = \mathcal{F}_2$.

Suppose $\overline{X} \in \mathcal{F}_1$ and $X := \overline{X}$ has the eigenvalue decomposition (5). Because $\mathrm{rank}(\overline{X}) \le r$, we have $\lambda_i = 0$ for $i \ge r+1$. Then $\overline{U} = P_1 P_1^T$ solves (17) by (8), and by (15)–(17) we have

$$\langle \overline{X}, \overline{U} \rangle = \sum_{i=1}^{r} \lambda_i = \mathrm{trace}(X) = n.$$

Therefore, $(\overline{X}, \overline{U})$ is a feasible point of (4), implying $\overline{X} \in \mathcal{F}_2$.

Conversely, let us assume $\overline{X} \in \mathcal{F}_2$. Then there exists a matrix $\overline{U} \in \mathcal{S}^n$ such that $(\overline{X}, \overline{U})$ is a feasible point of (4). Once again, let $X := \overline{X}$ have the eigenvalue decomposition (5). Then the optimal value of (17) is not bigger than the trace of $\overline{X}$, i.e.,

$$\sum_{i=1}^{r} \lambda_i \le \mathrm{trace}(\overline{X}) = n, \tag{18}$$

where the inequality holds because $\overline{X} \succeq 0$. We also note that $\langle \overline{X}, \overline{U} \rangle$ achieves its maximal value $n$:

$$\sum_{i=1}^{r} \lambda_i \ge \langle \overline{X}, \overline{U} \rangle = n. \tag{19}$$

The fact $X \succeq 0$, together with (18) and (19), implies $\lambda_i = 0$ for $i \ge r+1$. That is, $\mathrm{rank}(\overline{X}) \le r$, implying $\overline{X} \in \mathcal{F}_1$. The claim of the theorem follows easily from $\mathcal{F}_1 = \mathcal{F}_2$. ∎

**3. The algorithm and its convergence.** As mentioned in the introduction, the main computational burden of our algorithm is on the quadratic semidefinite programming subproblem (9), which we regard as the inner problem of our overall algorithm. In this section, we describe our algorithm, mainly focusing on the outer iterations, and analyze its convergence to a stationary point. The inner problem will be solved by our semismooth Newton method in section 4 to the accuracy specified by the algorithm.

**3.1. Description of the algorithm.** For given $U \in \mathcal{S}^n$ and $\mu, c \in \mathbb{R}$, define

$$L_{(U,\mu,c)}(X) := \frac{1}{2}\|X - C\|^2 - \mu\Big(\langle X, U\rangle - n\Big) + \frac{c}{2}\Big(\langle X, U\rangle - n\Big)^2.$$

The subproblem that our algorithm aims to solve in each iteration takes the following form:

$$\begin{array}{lll} \min\limits_{X \in \mathcal{S}^n} & L_{(U,\mu,c)}(X) & \\ \text{s.t.} & \mathrm{diag}(X) = e & (y), \\ & X \succeq 0, & (S), \end{array} \tag{20}$$

where $y \in \mathbb{R}^n$ and $S \in \mathcal{S}^n$ in brackets are the Lagrange multipliers of the corresponding constraints.

The KKT condition of (20) is

$$\begin{cases} R_p := \mathrm{diag}(X) - e = 0, \\ R_d := X - \Big(C + \mathrm{Diag}(y) + \mu U\Big) + c\Big(\langle X, U\rangle - n\Big)U - S = 0, \\ X \succeq 0,\ S \succeq 0,\ \langle X, S\rangle = 0. \end{cases} \tag{21}$$

It is well known (see [8]) that the complementarity condition (i.e., the last condition) in (21) is equivalent to

$$R_c := X - [X - S]_+ = 0,$$

where $[X]_+$ denotes the orthogonal projection of $X \in \mathcal{S}^n$ onto $\mathcal{S}^n_+$. An approximate solution to the subproblem (20) is often measured by

$$\|R_p\| + \|R_d\| + \|R_c\| \le \epsilon$$

for some small tolerance $\epsilon \ge 0$.

ALGORITHM 3.1.

**(S.0)** Let $\{\epsilon_k\} \downarrow 0$. Choose initial point $X^0 \in \mathcal{S}^n$; Lagrange multiplier $\mu_0$; penalty parameters $c_0 > 0$, $\rho > 1$, and $\tau > 0$. Conduct the eigenvalue decomposition (5) on $X := X^0$. Let $P^0 := P$, and let $P_1^0$ denote the submatrix consisting of the first $r$ columns of matrix $P^0$. Let $k := 0$.

**(S.1)** Let

$$U^k := P_1^k (P_1^k)^T.$$

**(S.2)** Find a triple $(X^{k+1}, y^{k+1}, S^{k+1})$ of the following subproblem:

$$\begin{array}{ll} \min_{X \in \mathcal{S}^n} & L_{(U^k, \mu_k, c_k)}(X) \\ \text{s.t.} & \mathrm{diag}(X) = e, \\ & X \succeq 0, \end{array} \tag{22}$$

such that $\|R_p\| + \|R_d\| + \|R_c\| \le \epsilon_k$.

**(S.3)** Conduct the eigenvalue decomposition (5) with $X := X^{k+1}$. Let $P^{k+1} := P$, and let $P_1^{k+1}$ denote the submatrix consisting of the first $r$ columns of matrix $P^{k+1}$.

**(S.4)** Update the Lagrange multiplier by

$$\mu_{k+1} := \max\left\{0,\ \mu_k - \Big(\langle X^{k+1}, U^k\rangle - n\Big)c_k\right\}.$$

Update the penalty parameter $c_k$ by

$$c_{k+1} := \max\left\{\rho c_k,\ |\mu_{k+1}|^{1+\tau}\right\}.$$

**(S.5)** Set $k := k + 1$ and go to (S.1).

We have a few comments on Algorithm 3.1.

(R1) We did not include any stopping criterion in Algorithm 3.1 because of the following two reasons. First, according to the KKT condition (11) of (4), we need the availability of the Lagrange multiplier $(\bar{y}, \bar{\mu}, \bar{\nu}, \overline{R})$ in order to verify whether $(\overline{X}, \overline{U})$ is a stationary point. Some (not all) of the multipliers may come from solving the subproblem (22) depending on which method is being used. Therefore, it would be difficult at this stage to give a meaningful stopping criterion. Second, the situation will become clear and simple when it comes to our implementation. A simple criterion is proposed in (53) and studied in Proposition 5.1.

(R2) In the standard augmented Lagrangian method (see, e.g., [22]), the Lagrange multiplier $\mu$ should be updated by the formula $\mu_{k+1} := \mu_k - (\langle X^{k+1}, U^k\rangle - n)c_k$, rather than taking its nonnegative part as adopted in (S.4). It is because

we have proved in Proposition 2.1 that the correct $\mu$ must be nonnegative. As a matter of fact, if subproblem (22) is solved to high accuracy, we always have $\langle X^{k+1}, U^k\rangle - n \le 0$ because $X^{k+1}$ is always a correlation matrix. Therefore, $\{\mu_k\}$ is nondecreasing. If $\mu_0 = 0$, the formula for (S.4) reduces to $\mu_{k+1} := \mu_k - (\langle X^{k+1}, U^k\rangle - n)c_k$. That the penalty parameter in the updating formula in (S.4) always grows faster than the Lagrange multiplier was recently introduced by Lu and Zhang [20] to deal with the feasibility issue at the limit in their augmented Lagrangian method.

(R3) We allow freedom to use any suitable algorithm to solve the subproblem (22) as long as the algorithm is able to produce a triple $(X^{k+1}, y^{k+1}, S^{k+1})$ that satisfies the required accuracy. $(y^{k+1}, S^{k+1})$ may not be required to be in an explicit form. In fact, Algorithm 3.1 needs only the availability of $X^{k+1}$. However, the information about $(y^{k+1}, S^{k+1})$ is important to our convergence analysis below.

(R4) The updating rule on $U$ (see (S.1) in Algorithm 3.1) makes use of the information of eigenvalue decomposition that is already available due to solving subproblem (22) from the last iteration. There may be other ways to update $U$. For example, the minimization problem (4) in $(X, U)$ can be cast into the framework of a nested minimization,[2] i.e., an outer minimization in $U$ and an inner minimization in $X$:

$$\min_{\substack{0\preceq U\preceq I\\ \langle I,U\rangle = r}} \quad \min_{\substack{X\succeq 0,\ \mathrm{diag}(X)=e\\ \langle X,U\rangle = n}} \frac{1}{2}\|X - C\|^2.$$

However, the inner problem may not be feasible for a given $U$. This prompted us to consider the augmented Lagrange problem in replace of the inner problem:

$$\phi(U) := \min_{X\succeq 0, \mathrm{diag}(X)=e} \frac{1}{2}\|X - C\|^2 - \mu(\langle X, U\rangle - n) + \frac{c}{2}(\langle X, U\rangle - n)^2.$$

The nested minimization now becomes

$$\min_{\substack{0\preceq U\preceq I\\ \langle I,U\rangle = r}} \quad \phi(U). \tag{23}$$

The optimal solution of (23) can then be used to update $U$ instead of our update rule (8). We tried various numerical schemes for (23); none of them gained a better performance than (8). However, this suggestion is very interesting and is more natural than (8), and hence it is worth further investigation.

**3.2. Convergence to stationary point.** First, note that $\|U^k\| = r$ for all $k = 0, \ldots$. In other words, $\{U^k\}$ is uniformly bounded. Moreover, the sequence $\{X^k\}$ is also bounded due to the boundedness of the feasible region of (22). If $\epsilon_k = 0$, then $\{(y^{k+1}, S^{k+1})\}$ is a unique Lagrange multiplier of (22) due to the constraint nondegeneracy in Lemma 2.2 and [2, Thm. 5.85]. Moreover, the practical stopping criterion (53) used in our implementation measures only some quantities on the sequence $\{(X^k, U^k)\}$. From this point of view, the assumption that there exists a converging subsequence in the following result is not very restrictive.

[2]We thank the referee for bringing this framework to our attention.

THEOREM 3.2. *Let $\{X^k\}$ be the sequence generated by Algorithm* 3.1. *Suppose that a subsequence $\{(X^{k+1}, y^{k+1}, S^{k+1})\}_{k\in K}$ converges to $(\overline{X}, \bar{y}, \overline{S})$ and $\{U^k\}_{k\in K}$ converges to $\overline{U}$. Then, the following statements hold:*

(i) $\langle \overline{X}, \overline{U}\rangle = n$.

(ii) $\{\mu_{k+1}\}_{k\in K}$ *is bounded.*

(iii) $(\overline{X}, \overline{U})$ *is a stationary point of* (4).

*Proof.* (i) By (S.3) in Algorithm 3.1, $\{(X^{k+1}, y^{k+1}, S^{k+1})\}$ satisfies the following conditions:

$$\left\| X^{k+1} - \Big(C + \mathrm{Diag}(y^{k+1}) + \mu_k U^k\Big) + c_k\Big(\langle X^{k+1}, U^k\rangle - n\Big)U^k - S^{k+1}\right\| \le \epsilon_k, \tag{24}$$

$$\|\mathrm{diag}(X^{k+1}) - e\| \le \epsilon_k, \tag{25}$$

$$\|X^{k+1} - [X^{k+1} - S^{k+1}]_+\| \le \epsilon_k. \tag{26}$$

Since $\{\epsilon_k\} \downarrow 0$, we follow from (25) and (26) that

$$\mathrm{diag}(\overline{X}) = e \qquad \text{and} \qquad \overline{X} - [\overline{X} - \overline{S}]_+ = 0. \tag{27}$$

By the formula for $c_k$ in (S.4) of Algorithm 3.1, we have $c_k \to \infty$. Dividing (24) by $c_k$ and taking limits on both sides yield

$$|\langle \overline{X}, \overline{U}\rangle - n| = 0, \tag{28}$$

where we used the fact that $c_k$ outgrows $\mu_k$ by at least a factor $|\mu_k|^\tau$.

(ii) To prove the boundedness of $\{\mu_{k+1}\}_{k\in K}$, we may assume, without loss of generality, that

$$\mu_{k+1} = \mu_k - c_k(\langle X^{k+1}, U^k\rangle - n) \ge 0 \qquad \forall\ k \in K.$$

Otherwise, $\mu_{k+1} \equiv 0$ for any $k \in K$, and hence $\{\mu_{k+1}\}_{k\in K}$ is bounded. It again follows from (24) that

$$\left\|X^{k+1} - \big(C + \mathrm{Diag}(y^{k+1})\big) - \mu_{k+1}U^k - S^{k+1}\right\| \le \epsilon_k. \tag{29}$$

Assume that $\{\mu_{k+1}\}_{k\in K}$ is unbounded. Dividing both sides of (29) by $\mu_{k+1}$ and taking limits yield

$$\|\overline{U}\| = 0,$$

contradicting $\|\overline{U}\| = r$. This establishes (ii).

(iii) Since $\{\mu_{k+1}\}_{k\in K}$ is bounded, we may assume without loss of generality that $\{\mu_{k+1}\}_{k\in K} \to \bar{\mu}$. Note that $\bar{\mu} \ge 0$. Taking limits on both sides of (29) gives

$$\overline{X} - (C + \mathrm{Diag}(\bar{y}) + \bar{\mu}\overline{U}) - \overline{S} = 0. \tag{30}$$

Recall that $P^{k+1}$ is the matrix from the eigenvalue decomposition (5) with $X := X^{k+1}$ and $P_1^{k+1}$ is the submatrix consisting of the first $r$ columns of $P^{k+1}$. Subsequencing if necessary, we may assume $\{P^{k+1}\}_{k\in K} \to \overline{P}$. Let $\overline{P}_1$ be the submatrix consisting of the first $r$ columns of $\overline{P}$.

We must have

$$\lim_{k\in K,\ k\to\infty} X^{k+1} = \overline{X} = \overline{P}\mathrm{Diag}(\bar{\lambda}_1, \ldots, \bar{\lambda}_n)\overline{P}^T \qquad \text{and} \qquad \overline{U} = \overline{P}_1\overline{P}_1^T,$$

where $\bar{\lambda}_i$, $i = 1, \ldots, n$ are eigenvalues of $\overline{X}$. Due to the continuity of eigenvalues, we must have $\bar{\lambda}_1 \geq \bar{\lambda}_2 \geq \cdots \geq \bar{\lambda}_n$. If follows from (27) and (28) that $\overline{X}$ is a correlation matrix, $\operatorname{rank}(\overline{X}) = r$, and

$$\overline{S} \succeq 0 \qquad \text{and} \qquad \langle \overline{X}, \overline{S} \rangle = 0. \tag{31}$$

Hence,

$$\bar{\lambda}_i = 0, \ i = r+1, \ldots, n.$$

Define the scalars

$$\bar{\nu} := 0, \ \tau_i := \bar{\mu}\bar{\lambda}_i, \ i = 1, \ldots, r,$$

and form the matrix

$$\overline{R} := \overline{P}\operatorname{Diag}(\tau_1, \ldots, \tau_r, 0, \ldots, 0)\overline{P}^T.$$

It is a matter of elementary linear algebra to check that $\overline{R} \succeq 0$ (because $\bar{\mu} \geq 0$) and

$$\langle I - \overline{U}, \overline{R} \rangle = 0, \ -\bar{\mu}\overline{X} - \bar{\nu} I + \overline{R} \succeq 0, \ \langle \overline{U}, \overline{R} \rangle = \bar{\mu} n + \bar{\nu} r. \tag{32}$$

Moreover, it follows from (30) and (31) that $\langle \overline{X}, \overline{S} \rangle = 0$ implies

$$\frac{1}{2}\|\overline{X} - C\|^2 + \frac{1}{2}(\|\overline{X}\| - \|C\|^2) - e^T\bar{y} - \bar{\mu} n = 0. \tag{33}$$

Putting the relationships (27), (28), and (30)–(33) together is sufficient to claim that $(\overline{X}, \overline{U})$, with $(\bar{y}, \bar{\mu}, \bar{\nu}, \overline{R})$, satisfies the KKT condition (11). Hence, $(\overline{X}, \overline{U})$ is a stationary point. □

**4. Solving the subproblem by semismooth Newton method.** The efficiency of Algorithm 3.1 depends on whether subproblem (22) can be efficiently solved. Being a standard quadratic semidefinite programming problem (QSDP), (22) could be solved by any QSDP solver such as the one proposed by Toh [36]. However, our numerical results showed that using a general QSDP solver may result in a quite slow Algorithm 3.1 due to the complicated objective function in (22). In this section, we develop a more efficient method called the *semismooth Newton method* to solve (22). It all starts from reformulating it as a least-square problem.

The idea is as follows. First, (9) is reformulated as a least-square problem. We then consider the Lagrangian dual of this least-square problem. The dual problem is unconstrained and convex. However, the objective function is only once continuously differentiable. Therefore, it is not possible to develop a classical Newton method for the dual problem. Nevertheless, we are able to develop a semismooth version of Newton's method, which is also quadratically convergent. Moreover, we can recover the solution of (9) from the solution of the dual problem.

**4.1. Least-square reformulation and its dual problem.** We introduce a new variable $z$ of one dimension by $z := \langle X, U \rangle - n$. The subproblem (20) can be equivalently stated as the least-square problem

$$\begin{array}{lll} \min\limits_{(X,z)\in\mathcal{S}^n\times\mathbb{R}} & \frac{1}{2}\|X - C\|^2 - \mu z + \frac{c}{2}z^2 & \\ \text{s.t.} & \operatorname{diag}(X) = e & (y), \\ & \langle X, U \rangle - z = n & (s), \\ & X \succeq 0, & \end{array} \tag{34}$$

where $y \in \mathbb{R}^n$ and $s \in \mathbb{R}$ in the brackets denote the Lagrange multipliers of the corresponding constraints.

Following the steps of calculations on the Lagrangian dual approach in [21] and [4], we can derive the Lagrangian dual problem of (34):

$$\min_{(y,s)\in\mathbb{R}^n\times\mathbb{R}} \theta(y,s) = \frac{1}{2}\|[C+\mathcal{A}^*(y,s)]_+\|^2 + \frac{1}{2c}(\mu - s)^2 - b^T(y^T,s)^T, \tag{35}$$

where the linear operator $\mathcal{A}: \mathcal{S}^n \mapsto \mathbb{R}^{n+1}$ and its adjoint operator $\mathcal{A}^*: \mathbb{R}^{n+1} \mapsto \mathcal{S}^n$ are given by

$$\mathcal{A}(X) := \begin{bmatrix} \mathrm{diag}(X) \\ \langle U, X\rangle \end{bmatrix} \quad \forall\ X \in \mathcal{S}^n;$$

$$\mathcal{A}^*(y,s) := \mathrm{Diag}(y) + sU \quad \forall\ (y,s) \in \mathbb{R}^n \times \mathbb{R}; \quad \text{and} \quad b := \begin{bmatrix} e \\ n \end{bmatrix}.$$

$\theta(\cdot,\cdot)$ is convex and continuously differentiable everywhere with

$$\nabla\theta(y,s) = \mathcal{A}\Big([C+\mathcal{A}^*(y,s)]_+\Big) - \frac{1}{c}\begin{bmatrix} 0_n \\ \mu - s \end{bmatrix} - b.$$

Moreover, if $(\bar{y},\bar{s}) \in \mathbb{R}^n \times \mathbb{R}$ is an optimal solution of (35), then $(\overline{X},\bar{z})$ given by

$$\overline{X} := [C+\mathcal{A}^*(\bar{y},\bar{s})]_+ \qquad \text{and} \qquad \bar{z} := \frac{1}{c}(\mu - \bar{s}) \tag{36}$$

is the optimal solution of (34). We have the following two remarks.

(R5) The function $\theta$ is coercive (i.e., $\theta \to +\infty$ as $\|(y,s)\| \to +\infty$) because the Slater condition holds for (34) and the linear constraints in (34) are linearly independent (since $U \neq I$).

(R6) The dual approach outlined above is classical and has been successfully applied to the nearest correlation matrix problem (3) (i.e., without the rank constraint) in [21, 4, 26, 3]. The minimization problem (35) is often regarded as the dual problem of (34). Instead of solving the subproblem (22) at iteration $X^k$ of Algorithm 3.1 (S.2), we try to solve its dual problem,

$$\min_{(y,s)\in\mathbb{R}^n\times\mathbb{R}} \theta_k(y,s), \tag{37}$$

where

$$\theta_k(y,s) := \frac{1}{2}\|[C+\mathcal{A}_k^*(y,s)]_+\|^2 + \frac{1}{2c_k}(\mu_k - s)^2 - b^T(y^T,s)^T \tag{38}$$

and $\mathcal{A}_k^*(y,s) := \mathrm{Diag}(y) + sU^k$. Suppose $(\bar{y},\bar{s})$ is an approximate solution of (37); the next iterate $X^{k+1}$ is taken to be $\overline{X}$ in (36). We can also construct $(y^{k+1},S^{k+1})$ from $X^{k+1}$ to meet the required accuracy in (S.2) of Algorithm 3.1 (see section 5.1 for a detailed argument). Below we are going to develop a fast algorithm to solve (37).

**4.2. Semismooth Newton method.** To continue from the above, we need to find a solution of (35) in order to calculate $\overline{X}$. Since $\theta$ is convex and coercive, it is equivalent to solve the optimality equation

$$\nabla\theta(y,s) = 0. \tag{39}$$

The equation is obviously nondifferentiable because of the projection operator $[\cdot]_+$. Therefore, the classical Newton method is not applicable.

However, it is a *strongly semismooth* equation [19, 28]. To see the strong semismoothness of $\nabla\theta(\cdot,\cdot)$, let us indicate the dependence of the operator $[X]_+$ on the positive semidefinite cone $\mathcal{S}^n_+$ by denoting $\Pi_{\mathcal{S}^n_+}(X) := [X]_+$ for any $X \in \mathcal{S}^n$. It is known that $\Pi_{\mathcal{S}^n_+}(\cdot)$ is strongly semismooth due to [34]. $\nabla\theta(\cdot,\cdot)$ is just a composite function of linear operators and $\Pi_{\mathcal{S}^n_+}(\cdot)$; it is therefore strongly semismooth.

The equation is very similar to that encountered in the nearest correlation matrix problem [21, 26]. The semismooth Newton method developed in [26] works extremely well with this kind of equation. We hence develop a semismooth Newton method for (39) in the form of

$$x^{k+1} = x^k - V_k^{-1}\nabla\theta(x^k), \qquad V_k \in \partial^2\theta(x^k), \tag{40}$$

where $x := (y, s) \in \mathbb{R}^n \times \mathbb{R}$, $x^k := (y^k, s_k) \in \mathbb{R}^n \times \mathbb{R}$ is the current iterate, and $\partial^2\theta(x^k)$ is the generalized Jacobian of $\theta$ at $x^k$ in the sense of Clarke [7].

For the method (40) to be quadratically convergent, we need to ensure that every element in $\partial^2\theta(\bar{x})$ is positive definite, where $\bar{x} := (\bar{y}, \bar{s})$. This positive definiteness property actually holds because of the following constraint nondegeneracy at $(\bar{y}, \bar{s})$.

PROPOSITION 4.1. *Let $(\overline{X}, \bar{z})$ denote the optimal solution of* (34). *Then constraint nondegeneracy holds at $(\overline{X}, \bar{z})$; i.e.,*

$$\overline{\mathcal{A}}\Big(\mathrm{lin}(T_{\mathcal{S}^n_+}(\overline{X})) \times \mathbb{R}\Big) = \mathbb{R}^{n+1}, \tag{41}$$

*where*

$$\overline{\mathcal{A}}(X, z) := \mathcal{A}(X) - \begin{bmatrix} 0_n \\ z \end{bmatrix} \qquad \forall\ (X, z) \in \mathcal{S}^n \times \mathbb{R}.$$

*Proof.* First we note that the definition of constraint nondegeneracy ([1] and [2, eq. 4.172]) when applied to the constraints in (34), is equivalent to (41). We prove that (41) must hold.

It is easy to see that (41) holds if and only if

$$\left\{\overline{\mathcal{A}}\Big(\mathrm{lin}(T_{\mathcal{S}^n_+}(\overline{X})) \times \mathbb{R}\Big)\right\}^{\perp} = \{0_{n+1}\}, \tag{42}$$

where the left-hand side of (42) denotes the subspace that is orthogonal to the subspace $\overline{\mathcal{A}}(\mathrm{lin}(T_{\mathcal{S}^n_+}(\overline{X})) \times \mathbb{R})$. Then, (42) holds if and only if the following implication holds:

$$\langle (y^T, s)^T, \overline{\mathcal{A}}(Z, t)\rangle = 0 \quad \forall\ (Z, t) \in \mathrm{lin}(T_{\mathcal{S}^n_+}(\overline{X})) \times \mathbb{R} \quad \Longrightarrow \quad y = 0,\ s = 0. \tag{43}$$

The left-hand side of (43) has the following expression:

$$\begin{aligned} 0 &= \langle \overline{\mathcal{A}}^*(y, s), (Z, t)\rangle \\ &= \langle \mathrm{Diag}(y) + sU, Z\rangle - st, \end{aligned} \tag{44}$$

where we used the relation

$$\overline{\mathcal{A}}^*(y, s) = \begin{bmatrix} \mathcal{A}^*(y, s) \\ -s \end{bmatrix}.$$

Let $Z = 0 \in \mathrm{lin}(T_{\mathcal{S}^n_+}(\overline{X}))$ in (43); we see from (44) that $st = 0$ for all $t \in \mathbb{R}$. This implies $s = 0$. Then, the left-hand side of (43) reduces to

$$\langle \mathrm{Diag}(y), Z\rangle = 0 \qquad \forall\ Z \in \mathrm{lin} T_{\mathcal{S}^n_+}(\overline{X}). \tag{45}$$

Since $\overline{X}$ is a correlation matrix, it follows from Lemma 2.2 that $\overline{X}$ is constraint nondegenerate. Putting (45) and (14) together, we get $y = 0$. This proves $y = 0$, $s = 0$. That is, the implication (43) is satisfied, and hence $(\overline{X}, \bar{z})$ is constraint nondegenerate. □

Proposition 4.1 may be regarded as a natural extension of Lemma 2.2. It basically says that if we add one more trace constraint to the correlation matrix set (13) with a new (one-dimensional) variable, any feasible point of this new set is still constraint nondegenerate. With this key property in hand, we are able to prove the positive definiteness of $\partial^2\theta(\bar{y}, \bar{x})$ by following a proof similar to that of [26, Prop. 3.6]. One also needs to characterize the structure of $\partial^2\theta(\bar{y}, \bar{x})$, which can also be done as in [26, Lem. 3.5]. We omit the details.

Now we are ready to state the semismooth Newton method for (39). This method has also been used in [26, 40] for similar semismooth equations.

Algorithm 4.2 (semismooth Newton Method).

**(S.0)** Given $x^0 \in \mathbb{R}^{n+1}$, $\eta \in (0,1)$, $\sigma \in (0,1)$, $\kappa_1 \in (0,1)$, $\kappa_2 \in (1,\infty)$, $\kappa_3 \in (1,\infty)$, and $\delta \in (0,1)$, let $j := 0$.

**(S.1)** Select an element $V_j \in \partial^2\theta(x^j)$, compute $t_j := \min\{\kappa_1, \kappa_2\|\nabla\theta(x^j)\|\}$, and apply the CG method [14] starting with the zero vector as the initial search direction to

$$(V_j + t_j I)\Delta x = -\nabla\theta(X^j) \tag{46}$$

to find a search direction $\Delta x^j$ such that

$$\|\nabla\theta(x^j) + (V_j + t_j I)\Delta x^j\| \le \eta_j \|\nabla\theta(x^j)\|\,, \tag{47}$$

where $\eta_j := \min\{\eta, \kappa_3\|\nabla\theta(x^j)\|\}$.

**(S.2)** Let $l_j$ be the smallest nonnegative integer $l$ such that

$$\theta(x^j + \delta^l \Delta x^j) - \theta(x^j) \le \sigma\, \delta^l \left\langle \nabla\theta(x^j), \Delta x^j \right\rangle .$$

Set $\tau_j := \delta^{l_j}$ and $x^{j+1} := x^j + \tau_j \Delta x^j$.

**(S.3)** Replace $j$ by $j + 1$ and go to (S.1).

The convergence analysis of this algorithm can be conducted in a way similar to that in [26, Thm. 5.3] or [40, Thm. 3.5], using the coerciveness of $\theta$ and the positive definiteness of $V \in \partial^2\theta(\bar{x})$. We state the convergence result in the next theorem without giving a detailed proof.

Theorem 4.3. *Suppose that in Algorithm* 4.2, $\nabla\theta(x^j) \neq 0$ *for all* $j \ge 0$. *Then Algorithm* 4.2 *is well defined, and the generated iteration sequence* $\{x^j\}$ *converges quadratically to the unique solution* $\bar{x}$ *of problem* (35).

**5. Numerical results.** We may regard Algorithm 3.1 as our outer algorithm and Algorithm 4.2 as the inner algorithm. We have to address whether the inner algorithm is able to provide a good enough approximate $(X^{k+1}, y^{k+1}, S^{k+1})$, as requested in (S.2) of Algorthim 3.1. This is related to the stopping criteria used in both the outer and inner algorithms.

**5.1. Stopping criteria.** Since $\theta$ is convex and the optimization problem (35) is unconstrained, an obvious stopping criterion in the inner problem is (recall $\bar{x} := (\bar{y}, \bar{s})$)

$$\|\nabla\theta(\bar{x})\| \le \mathtt{tol}_{\mathrm{in}}, \tag{48}$$

where $\mathtt{tol}_{\mathrm{in}} > 0$ is a given tolerance and $\bar{x}$ is the final iterate produced by Algorithm 4.2. For brevity, we omit the superscript $k$ of the iteration used by the outer algorithm.

For the convergence property (Theorem 3.2) of Algorithm 3.1 to be valid, we have to verify that we can find an approximate $(X, y, S)$ from $(\overline{X}, \bar{z})$ as requested in (S.2) of Algorithm 3.1. Denote $\epsilon := \mathtt{tol}_{\mathrm{in}}$. Then, (48) is equivalent to

$$\|\mathrm{diag}(\overline{X}) - e\| \le \epsilon \qquad \text{and} \qquad |\langle U, \overline{X}\rangle - n - (\mu - \bar{s})/c| \le \epsilon. \tag{49}$$

Let

$$\overline{S} := \overline{X} - (C + \mathrm{Diag}(\bar{y}) + \bar{s}U). \tag{50}$$

From the definition of $\overline{X}$ in (36), we must have

$$\overline{S} \succeq 0 \qquad \text{and} \qquad \langle \overline{X}, \overline{S}\rangle = 0. \tag{51}$$

It also follows from (50) that

$$\begin{aligned} &\|\overline{X} - (C + \mathrm{Diag}(\bar{y}) + \mu U) + c(\langle U, \overline{X}\rangle - n)U - \overline{S}\| \\ &\quad = \|\overline{X} - (C + \mathrm{Diag}(\bar{y}) + \bar{s}U) + c(\langle U, \overline{X}\rangle - n + ((\bar{s} - \mu)/c)U) - \overline{S}\| \\ &\quad = c|\langle U, \overline{X}\rangle - n + (\bar{s} - \mu)/c|\|U\| \le cr\epsilon. \end{aligned} \tag{52}$$

The inequality used (49) and $\|U\| = r$.

Putting (49), (51), and (52) together, we get that $(\overline{X}, \bar{y}, \overline{S})$ is an approximate solution of (20) satisfying $\|R_p\| + \|R_d\| + \|R_c\| = O(\epsilon)$. Therefore, at each iteration, Algorithm 4.2 provides an approximate solution to (22) as requested in (S.2) of Algorithm 3.1.

Now we address the outer stopping criterion, which is to test whether the current iterate $(X^k, U^k)$ satisfies

$$\left|\sum_{i=1}^{r} \lambda_i^k - n\right| \le \mathtt{tol}_{\mathrm{rank}} \qquad \text{and} \qquad |\langle X^k, U^k - U^{k-1}\rangle| \le \mathtt{tol}_{\mathrm{eig}}, \tag{53}$$

where $\lambda_i^k$, $i = 1, \ldots, r$, are the first $r$ largest eigenvalues of $X^k$, $\mathtt{tol}_{\mathrm{rank}} > 0$ is a given tolerance that controls the first $r$ leading eigenvalues, and $\mathtt{tol}_{\mathrm{eig}} > 0$ is a given tolerance that somewhat measures the closeness between $U^k$ and $U^{k-1}$.

Given starting point $X^0$, set $U^{-1} := U^0$ (recall $k$ in $U^k$ is the iteration index). According to Theorem 3.2(i), $|\sum_{i=1}^{r} \lambda_i^k - n| \to 0$ at least on a subsequence of $\{X^k\}_{k\in K}$. Moreover, we have the following property.

PROPOSITION 5.1. *Suppose* $\mathtt{tol}_{\mathrm{in}} = \mathtt{tol}_{\mathrm{rank}} = \mathtt{tol}_{\mathrm{eig}} = 0$. *If* $(X^k, U^k)$ *satisfies* (53), *then* $(X^k, U^k)$ *is a stationary point of* (4).

*Proof.* If $k = 0$, the stopping criterion (53) means $\mathrm{rank}(X^0) = r$ as $X^0$ is the nearest correlation matrix. Therefore, $X^0$ is the global solution of the low rank problem (1), and $(X^0, U^0)$ is a global solution of (4). It must be a stationary point.

Now suppose $k \geq 1$ and suppose $\bar{x} := (\bar{y}, \bar{s})$ is the final iterate produced by the semismooth Newton method applied to the dual problem (35) of the subproblem (34) at iteration $(k-1)$ (i.e., $U := U^{k-1}$, $\mu := \mu_{k-1}$, and $c := c_{k-1}$). Then,

$$X^k = \overline{X} = [C + \mathrm{Diag}(\bar{y}) + \bar{s}U^{k-1}]_+,$$

and $U^k$ is formed as in (S.1) of Algorithm 3.1. Consequently,

$$\langle X^k, U^k \rangle = \sum_{i=1}^{r} \lambda_i^k.$$

Since $\mathtt{tol}_{\mathrm{in}} = 0$, we have $\nabla\theta(\bar{x}) = 0$. Let $\overline{S}$ be defined as in (50). We then must have

$$\begin{cases} \mathrm{diag}(\overline{X}) - e = 0, \\ \langle U^{k-1}, \overline{X} \rangle - n - (\mu_{k-1} - \bar{s})/c = 0, \\ \overline{X} \succeq 0,\ \overline{S} \succeq 0, \quad \langle \overline{X}, \overline{S} \rangle = 0. \end{cases} \tag{54}$$

System (54) together with the fact that $\sum_{i=1}^{r} \lambda_i^k = n$ means that $\overline{X}$ is a correlation matrix. Therefore, we have $\langle \overline{X}, U^k \rangle = n$. With $\mathtt{tol}_{\mathrm{eig}} = 0$, there is

$$\langle \overline{X}, U^{k-1} \rangle = n. \tag{55}$$

It follows from the second equation in (54) that $\mu_{k-1} = \bar{s}$. Denote $\bar{\mu} := \mu_{k-1}$ and $\overline{U} := U^k$. It follows from the definition of $\overline{S}$ in (50) that

$$\begin{aligned} &\overline{X} - (C + \mathrm{Diag}(\bar{y}) + \bar{\mu}\overline{U}) + c(\langle \overline{X}, \overline{U} \rangle - n)\overline{U} - \overline{S} \\ &\quad = \overline{X} - (C + \mathrm{Diag}(\bar{y}) + \bar{\mu}\overline{U}) - \overline{S} \\ &\quad = 0. \end{aligned} \tag{56}$$

Putting together (54), (55), and (56) and repeating the proof of Theorem 3.2(iii), it is easy to see that $(\overline{X}, \overline{U}) = (X^k, U^k)$ is a stationary point. ∎

**5.2. Modified principal component analysis as final output.** We address the output solution $C_r$ by our method. Suppose that $X^f$ is the final iterate upon satisfying the stopping criterion (53), which means that the sum of the $r$ leading eigenvalues of $X^f$ is very close to $n$ and the remaining $(n-r)$ eigenvalues of $X^f$ are very small. However, they are not zero. Hence, $X^f$ is not a true low-rank matrix in the mathematical sense. We apply the modified principal component analysis (PCA) to $X^f$ to output $C_r$. Let $X^f$ have the eigenvalue decomposition (5) with $X := X^f$. Let $\Lambda_r := \mathrm{Diag}(\lambda_1, \ldots, \lambda_r)$. Define

$$\{X_{\mathrm{pca}}\}_{i\cdot} := \frac{\mathbf{z}_i}{\|\mathbf{z}_i\|} \qquad \text{with} \qquad \mathbf{z_i} := \{P_1 \Lambda_r^{1/2}\}_{i\cdot}, \quad i = 1, \ldots, n, \tag{57}$$

where $\{Y\}_{i\cdot}$ denotes the $i$th row of a matrix $Y$. Then, the output $C_r$ is given by

$$C_r := (X_{\mathrm{pca}})(X_{\mathrm{pca}})^T. \tag{58}$$

The modified PCA (57)–(58) is due to Flury [9], and a description of it in finance-related articles can be found in [17, 25]. $C_r$ is guaranteed to be a rank $r$ correlation matrix. If $\lambda_r = 0$, we consider only eigenvalues up to $\lambda_{r-1}$. Then, the output would be $C_{r-1}$.

The PCA modification may increase the distance from $C$. The next result, in the spirit of [3, Lem. 3.2], provides a bound on the increase.

PROPOSITION 5.2. *Let $X^f$ be the final iterate satisfying the outer stopping criterion* (53). *Then,*

$$\|C_r - C\| \le \|X^f - C\| + n\left(\mathtt{tol} + \frac{\mathtt{tol}}{1-\mathtt{tol}}\right),$$

*where* $\mathtt{tol} := \mathtt{tol}_{\rm in} + \mathtt{tol}_{\rm rank}$.

*Proof.* Let $X^f$ have the eigenvalue decomposition (5) with $X := X^f$. We note that

$$X^f = \Pi_{\mathcal{S}^n_+}(C + \mathcal{A}^*(\bar{x})) \qquad \text{and} \qquad \sum_{\ell=r+1}^{n} \lambda_\ell \le \mathtt{tol}_{\rm rank}, \tag{59}$$

for some $\bar{x}$ satisfying $\|\nabla\theta(\bar{x})\| \le \mathtt{tol}_{\rm in}$. By using the formulation of $\nabla\theta(\cdot)$, we have

$$\mathrm{diag}(X^f) = \{\nabla\theta(\bar{x})\}_{1:n} + e,$$

where $\{\nabla\theta(\bar{x})\}_{1:n}$ is the vector of the first $n$ components of $\nabla\theta(\bar{x})$. Therefore, the diagonal elements of $X^f$ have the following bound:

$$1 - \mathtt{tol}_{\rm in} \le (X^f)_{ii} \le 1 + \mathtt{tol}_{\rm in}, \qquad i = 1, \ldots, n. \tag{60}$$

Through the eigenvalue decomposition (5), we have

$$X^f_{ii} = \sum_{\ell=1}^{n} \lambda_\ell P_{i\ell}^2, \qquad i = 1, \ldots, n.$$

Let

$$\bar{\mathbf{z}}_i := \{P\Lambda^{1/2}\}_{i\cdot}, \quad i = 1, \ldots, n.$$

Then,

$$X^f_{ii} = \|\bar{\mathbf{z}}_i\|^2 = \|\mathbf{z}_i\|^2 + \sum_{\ell=r+1}^{n} \lambda_\ell P_{i\ell}^2 \le \|\mathbf{z}_i\|^2 + \sum_{\ell=r+1}^{n} \lambda_\ell,$$

which implies by (60) and (59) that

$$\|\mathbf{z}_i\|^2 \ge 1 - (\mathtt{tol}_{\rm in} + \mathtt{tol}_{\rm rank}) = 1 - \mathtt{tol}. \tag{61}$$

Furthermore,

$$\begin{aligned} X^f_{ij} - (C_r)_{ij} &= \sum_{\ell=1}^{n} \lambda_\ell P_{i\ell} P_{j\ell} - \frac{1}{\|\mathbf{z}_i\|\|\mathbf{z}_j\|} \sum_{\ell=1}^{r} \lambda_\ell P_{i\ell} P_{j\ell} \\ &= \sum_{\ell=r+1}^{n} \lambda_\ell P_{i\ell} P_{j\ell} + \left(\sum_{\ell=1}^{r} \lambda_\ell P_{i\ell} P_{j\ell}\right)\left(1 - \frac{1}{\|\mathbf{z}_i\|\|\mathbf{z}_j\|}\right). \end{aligned} \tag{62}$$

By the Cauchy–Schwarz inequality, there is

$$\begin{aligned} \left|\sum_{\ell=1}^{r} \lambda_\ell P_{i\ell} P_{j\ell}\right| &\le \left(\sum_{\ell=1}^{r} \lambda_\ell P_{i\ell}^2\right)^{1/2} \left(\sum_{\ell=1}^{r} \lambda_\ell P_{j\ell}^2\right)^{1/2} \\ &\le (1 + \mathtt{tol}_{\rm in})^{1/2}(1 + \mathrm{tol}_{\rm in})^{1/2} = 1 + \mathtt{tol}_{\rm in}, \end{aligned}$$

TABLE 1
*Statistics of $\gamma_i$, $i = 1, \ldots, 4$.*

| | $\gamma_1$ | $\gamma_2$ | $\gamma_3$ | $\gamma_4$ |
|---|---|---|---|---|
| Estimate | 0.000 | 0.480 | 1.511 | 0.186 |
| Standard error | - | 0.099 | 0.289 | 0.127 |

and

$$\left| \sum_{\ell=r+1}^{n} \lambda_\ell P_{i\ell} P_{j\ell} \right| \leq \sum_{\ell=r+1}^{n} \lambda_\ell \leq \texttt{tol}_{\text{rank}}.$$

It follows from (62) and (61) that

$$\begin{aligned} |X^f_{ij} - (C_r)_{ij}| &\leq \texttt{tol}_{\text{rank}} + (1 + \texttt{tol}_{\text{in}}) \left( \frac{1}{\|\mathbf{z}_i\| \|\mathbf{z}_j\|} - 1 \right) \\ &\leq \text{tol} \left( 1 + \frac{1}{1 - \texttt{tol}} \right). \end{aligned}$$

We then have

$$\|C_r - C\| \leq \|X^f - C\| + \|X^f - C_r\| \leq \|X^f - C\| + n \left( \texttt{tol} + \frac{\texttt{tol}}{1 - \texttt{tol}} \right). \qquad \square$$

**5.3. Test problems.** We are going to test the following problems, collected from various sources:

E1 $C_{ij}$ is the correlation between forward rates $i$ and $j$ and is given by $C_{ij} = 0.5 + (1 - 0.5)\exp(-0.05|i - j|)$ [5].

E2 $C = E + 0.05R$, where $E$ and $R$ are generated by `d = 10.^(4*[-1:1/(n-1):0])`, `E = gallery('randcorr',n*d/sum(d))`, `R = 2*rand(n) - 1`, `R = triu(R) + triu(R,1)'` [40].

E3 The matrix $C$ is the $387 \times 387$ one-day correlation matrix (as of Oct. 10, 2008) from the lagged datasets of RiskMetrics (the example can be obtained from the authors).

E4 $C_{ij}$ is also the correlation between forward rates $i$ and $j$ which has the following form:

$$C_{ij} = \text{LongCorr} + (1 - \text{LongCorr}) \exp(\kappa |t_i - t_j|),$$

where LongCorr $= 0.6$, $\kappa = -0.1$, $t_i = i$ [31, sect. 9.3].

E5 The random "interest rate" correlation matrix $C$ is given by

$$C_{ij} = \exp\left( -\gamma_1 |t_i - t_j| - \frac{\gamma_2 |t_i - t_j|}{\max(t_i, t_j)^{\gamma_3}} - \gamma_4 |\sqrt{t_i} - \sqrt{t_j}| \right),$$

with $\gamma_i > 0$, $i = 1, \ldots, 4$, and $t_i$ denoting the expiry time of rate $i$ and $t_i = i$. C is generated by randomizing the $\gamma$ parameters, with means and standard errors given by Table 1 with $\gamma_1$, $\gamma_2$, $\gamma_4$ capped at zero [25] .

E6 $n = 100$. $C$ is generated by the MATLAB `gallery('randcorr',n)`. If $|C_{ij}| \leq 0.01$, let $C_{ij} = 0$. Besides diagonal constraints, we require $X_{ij} = 0$ if $C_{ij} \leq 0.01$ [33].

TABLE 2
*Comparison of different starting points.*

| | $t$ | | | $f$ | | | iter | | |
|---|---|---|---|---|---|---|---|---|---|
| | (a) | (b) | (c) | (a) | (b) | (c) | (a) | (b) | (c) |
| E1n100r5 | 0.4 | 0.5 | 0.7 | 5.47 | 5.95 | 13.31 | 3 | 4 | 5 |
| E1n100r10 | 0.4 | 0.5 | 0.6 | 1.93 | 2.02 | 4.70 | 4 | 4 | 5 |
| E2n100r5 | 0.7 | 0.5 | 0.6 | 32.01 | 32.11 | 45.28 | 4 | 4 | 4 |
| E2n100r10 | 0.7 | 0.5 | 0.7 | 15.51 | 15.56 | 26.28 | 4 | 4 | 5 |
| E3n387r5 | 12.5 | 15.5 | 14.1 | 61.57 | 62.15 | 87.68 | 5 | 6 | 7 |
| E3n387r10 | 11.5 | 12.1 | 13.0 | 24.12 | 24.44 | 55.35 | 5 | 5 | 7 |
| E4n100r5 | 0.4 | 0.5 | 0.6 | 7.67 | 7.89 | 15.78 | 4 | 4 | 5 |
| E4n100r10 | 0.4 | 0.4 | 0.6 | 2.97 | 3.05 | 5.89 | 4 | 4 | 5 |
| E5n100r5 | 0.5 | 0.4 | 0.8 | 4.18 | 4.28 | 8.40 | 4 | 4 | 6 |
| E5n100r10 | 0.4 | 0.5 | 0.6 | 1.51 | 1.57 | 3.11 | 4 | 4 | 5 |

**5.4. Numerical tests.** In this section, we first describe the general parameter setting in our numerical test, followed by thorough comparisons with other existing methods.

(a) *General parameter setting.* Algorithm 3.1 was coded in MATLAB and was run on a desktop with Intel Core 2 of 3.40GHz and 3.39GHz CPU and 1.00Gb of RAM.

The parameters in Algorithm 3.1 were set as follows: $c_0 = 1$, $\mu_0 = 0$, $\rho = 1.2658$, $\tau = 0$. For the starting point $X^0$, we first get the nearest correlation matrix $X_{ncm}$ of problem (3), which can be solved by the semismooth Newton method of [26]; then we use modified PCA to $X_{ncm}$ to get $X^0$. As for the stopping test, we use

$$\begin{cases} \texttt{tol}_{\text{rank}} = \texttt{atol}_1 + \texttt{rtol}_1 \cdot |\sum_{i=1}^{r} \lambda_i^1 - n|, \\ \texttt{tol}_{\text{eig}} \ = \texttt{atol}_2 + \texttt{rtol}_2 \cdot |\langle X^1, U^1 - U^0 \rangle|, \end{cases}$$

where $\texttt{atol}_1 = 2 \times 10^{-4}$, $\texttt{rtol}_1 = 5 \times 10^{-4}$, $\texttt{atol}_2 = 0.03$, and $\texttt{rtol}_2 = 10^{-3}$. The stopping criterion above was motivated by [18, sect. 1.5] for solving nonlinear equations.

The parameters in Algorithm 4.2 were set as follows: $\kappa_1 = 10^{-10}$, $\kappa_2 = 1$, $\kappa_3 = 10^4$, $\eta = 10^{-2}$, $\delta = 0.5$, $\sigma = 10^{-4}$, and $\texttt{tol}_{\text{in}} = \texttt{atol}_3 \|\nabla\theta(y^0)\| + 10^{-8}$ with $\texttt{atol}_3 \in [10^{-8}, 10^{-0.8}]$.

In our reported results, we denote our method as `SemiNewton` and the objective function value by $f = \|C_r - C\|$, where $C_r$ is the output solution by our method. iter denotes the number of iterations in Algorithm 3.1, and $t$ is the CPU time in seconds. We name our test problems in the following way. For example, E2n100r20 means E2 with $n = 100$ and $r = 20$.

(b) *Comparison of different starting points.* As for the starting point, we would like to point out that different starting points may lead to different local minima. To show the role that starting points play, we compare three choices: (a) $X^0 = X_{ncm}$, (b) $X^0 = 0$, and (c) $X^0 = (X + X^T)/2$ with $X = \texttt{rand}(n, n)$, and $X^0$ is further modified by PCA. The results in Table 2 show that the three choices work very well. However, the comparison of function values implies that the solutions of (a) and (b) seem to be close to the same local minima, whereas the solution of (c) lies near another local minimum, whose function value is significantly higher. Therefore, a good starting point is very important to our algorithm. Here, we use $X_{ncm}$, which overall performs better than others.

TABLE 3
*Comparison between $\|X^f - C\|$ and $\|C_r - C\|$ on E1.*

| E1n100 | | | E1n500 | | |
|---|---|---|---|---|---|
| $r$ | $\|X^f - C\|$ | $\|C_r - C\|$ | $r$ | $\|X^f - C\|$ | $\|C_r - C\|$ |
| 2 | 19.119040 | 19.119040 | 10 | 38.687960 | 38.687956 |
| 10 | 1.934012 | 1.933997 | 20 | 15.708086 | 15.708085 |
| 20 | 0.671405 | 0.671397 | 50 | 4.139404 | 4.139394 |
| 30 | 0.361528 | 0.361463 | 80 | 2.049923 | 2.049922 |

TABLE 4
*Comparison of function values for each iteration between* `SemiNewton` *and* `mSemiNewton`.

| | $k$ | 0 | 1 | 2 | 3 | 4 | 5 |
|---|---|---|---|---|---|---|---|
| E1n100r5 | `SemiNewton` | 8.417 | 3.913 | 5.354 | 5.474 | | |
| | `mSemiNewton` | 8.417 | 5.647 | | | | |
| E2n100r5 | `SemiNewton` | 32.557 | 24.090 | 32.164 | 32.198 | 32.187 | |
| | `mSemiNewton` | 32.557 | 32.244 | 32.201 | | | |
| E3n387r15 | `SemiNewton` | 15.440 | 7.967 | 11.181 | 11.726 | 11.955 | 11.940 |
| | `mSemiNewton` | 15.440 | 12.616 | 12.069 | 11.971 | | |
| E4n100r5 | `SemiNewton` | 12.632 | 5.606 | 7.612 | 7.672 | 7.669 | |
| | `mSemiNewton` | 12.632 | 7.932 | | | | |
| E5n100r5 | `SemiNewton` | 7.618 | 4.057 | 5.528 | 5.687 | 5.683 | |
| | `mSemiNewton` | 7.618 | 5.807 | | | | |

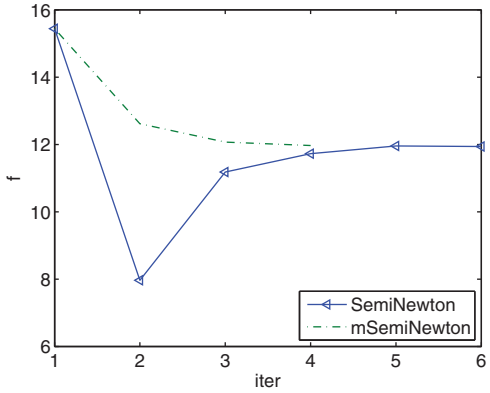


FIG. 1. E3n387r15: *Function values during the iterations.*

(c) *Comparison with modified PCA.* To give an idea of the difference between $\|C_r - C\|$ and $\|X^f - C\|$ estimated in Proposition 5.2, we report the typical results on E1 in Table 3, where one can find that the modified PCA (`mPCA`) in the final iteration brings little change in the distance from $C$. Therefore, in our following results, we report only $f = \|C_r - C\|$. For further investigation of the role that `mPCA` plays in `SemiNewton`, we compare `SemiNewton` with itself with `mPCA` being applied to each iterate $X^k$ (denoted as `mSemiNewton`). For each example, we report the objective function values during the iteration, and the results are listed in Table 4. Typical results on E3n387r15 are plotted in Figure 1. One can find that given the same starting points, `mSemiNewton` terminates earlier than `SemiNewton`, indicating that `mPCA` does accelerate the speed of the method; however, this is at the price of returning a slightly higher function value.

TABLE 5
*Comparison:* `Major`, `TPM`, *and* `SemiNewton` *on* $E1$.

| | $t$ | | | $f$ | | | iter | | |
|---|---|---|---|---|---|---|---|---|---|
| $r$ | `Major` | `TPM` | `SemiNewton` | `Major` | `TPM` | `SemiNewton` | `Major` | `TPM` | `SemiNewton` |
| | | | | $n = 10$ | | | | | |
| 2 | 0.06 | 0.03 | 0.03 | 2.77E-01 | 2.77E-01 | 2.77E-01 | 30 | 42 | 3 |
| 4 | 0.14 | 0.16 | 0.06 | 8.33E-02 | 8.32E-02 | 8.32E-02 | 97 | 163 | 3 |
| 6 | 0.22 | 0.79 | 0.01 | 4.16E-02 | 4.15E-02 | 4.15E-02 | 142 | 433 | 2 |
| 8 | 0.30 | 1.95 | 0.01 | 2.17E-02 | 2.19E-02 | 2.16E-02 | 180 | 569 | 2 |
| | | | | $n = 20$ | | | | | |
| 2 | 0.11 | 0.04 | 0.06 | 1.08E+00 | 1.08E+00 | 1.08E+00 | 20 | 48 | 3 |
| 4 | 0.30 | 0.30 | 0.03 | 3.38E-01 | 3.38E-01 | 3.38E-01 | 56 | 260 | 3 |
| 6 | 0.53 | 2.17 | 0.06 | 1.75E-01 | 1.75E-01 | 1.75E-01 | 104 | 985 | 2 |
| 8 | 0.80 | 6.56 | 0.03 | 1.10E-01 | 1.10E-01 | 1.10E-01 | 158 | 1507 | 2 |
| | | | | $n = 50$ | | | | | |
| 2 | 0.17 | 0.05 | 0.16 | 5.97E+00 | 5.97E+00 | 5.97E+00 | 13 | 59 | 3 |
| 4 | 0.34 | 1.48 | 0.14 | 2.05E+00 | 2.05E+00 | 2.05E+00 | 27 | 680 | 3 |
| 6 | 0.66 | 8.89 | 0.19 | 1.09E+00 | 1.09E+00 | 1.09E+00 | 51 | 1647 | 4 |
| 8 | 1.05 | 28.40 | 0.14 | 6.93E-01 | 6.92E-01 | 6.92E-01 | 79 | 2154 | 3 |
| | | | | $n = 100$ | | | | | |
| 2 | 0.38 | 0.12 | 0.44 | 1.91E+01 | 1.91E+01 | 1.91E+01 | 13 | 74 | 4 |
| 4 | 0.42 | 10.02 | 0.47 | 7.60E+00 | 7.60E+00 | 7.60E+00 | 17 | 1112 | 4 |
| 6 | 0.78 | 77.31 | 0.42 | 4.19E+00 | 4.19E+00 | 4.19E+00 | 29 | 3247 | 4 |
| 8 | 1.19 | 210.30 | 0.36 | 2.72E+00 | 2.72E+00 | 2.72E+00 | 44 | 4711 | 3 |

(d) *Comparison with trigonometric parameterization method.* One of the popular methods in practice is the trigonometric parameterization method (`TPM`). We choose the version of `TPM` in [29, sect. 9], where a geometric interpretation is given and the number of variables involved is further reduced to $l := (r-1)(n-\frac{r}{2})$. The problem is formulated as the following unconstrained optimization problem:

$$\min_{\Theta \in \mathbb{R}^l} F(\Theta) := \frac{1}{2}\|B(\Theta)B(\Theta)^T - C\|^2,$$

where $B(\Theta) \in \mathbb{R}^{n\times r}$ is given by (18) in [29]. We solve it by the built-in function `fminunc` in MATLAB. The starting point is chosen as $\Theta_0 = [0, -1, 2, -3, \ldots]$
$\times \frac{\pi}{2(l-1)} \in \mathbb{R}^l$, which leads to overall better performance than that of other choices. An analytical gradient is provided. We use a quasi-Newton method (`BFGS` Hessian update). Specifically, the parameters in "options" of `fminunc` are set as follows:

options = optimset("LargeScale," "off," "GradObj," "on," "MaxIter,"10000,
"HessUpdate," "bfgs").

We tested many examples, and our observation is consistent. We list just one of the tested examples (E1) in Table 5, where $f := \|B(\Theta)B(\Theta)^T - C\|$. One can find that when the dimension of the problem is not big ($n \le 50$) and $r$ is small, `TPM` is competitive with `Major` and `SemiNewton`. However, when $n$ is larger, even for small $r$, both `Major` and `SemiNewton` significantly outperform `TPM`. This is also consistent with the general view on `TPM` (see, e.g., [25, 12]). Therefore we do not include any further results on `TPM`.

TABLE 6
*Comparison between* `SemiNewton` *and* `Major`.

| | $t$ | | $f$ | | iter | |
|---|---|---|---|---|---|---|
| $r$ | `SemiNewton` | `Major` | `SemiNewton` | `Major` | `SemiNewton` | `Major` |
| | E2n100 | | | | | |
| 1 | 1.6 | 0.2 | 95.98 | 95.63 | 7 | 3 |
| 2 | 0.5 | 0.2 | 63.24 | 63.12 | 3 | 6 |
| 5 | 0.6 | 0.3 | 32.09 | 32.03 | 4 | 8 |
| 10 | 0.6 | 0.3 | 15.54 | 15.54 | 4 | 6 |
| 20 | 0.7 | 0.4 | 5.41 | 5.41 | 5 | 9 |
| 30 | 0.5 | 0.7 | 2.52 | 2.52 | 3 | 14 |
| 40 | 0.4 | 0.9 | 1.68 | 1.68 | 2 | 20 |
| 60 | 0.4 | 3.1 | 1.39 | 1.39 | 3 | 43 |
| | E3n387 | | | | | |
| 1 | 30.3 | 1.5 | 311.41 | 310.64 | 10 | 3 |
| 2 | 14.6 | 1.7 | 162.98 | 162.35 | 6 | 5 |
| 5 | 12.0 | 4.4 | 61.57 | 61.13 | 5 | 28 |
| 10 | 11.1 | 10.1 | 24.12 | 24.02 | 5 | 64 |
| 20 | 8.1 | 19.8 | 6.09 | 6.08 | 4 | 99 |
| 30 | 5.3 | 30.8 | 2.03 | 2.03 | 3 | 118 |
| 40 | 7.2 | 59.0 | 0.78 | 0.78 | 4 | 198 |
| 60 | 5.9 | 186.0 | 0.16 | 0.16 | 3 | 388 |
| | E4n100 | | | | | |
| 1 | 0.4 | 0.1 | 34.29 | 34.29 | 4 | 1 |
| 2 | 0.4 | 0.6 | 20.80 | 20.71 | 4 | 22 |
| 5 | 0.4 | 0.5 | 7.67 | 7.67 | 4 | 18 |
| 10 | 0.4 | 1.2 | 2.97 | 2.97 | 4 | 44 |
| 20 | 0.4 | 4.1 | 1.06 | 1.06 | 4 | 122 |
| 30 | 0.3 | 8.5 | 0.58 | 0.58 | 3 | 211 |
| 40 | 0.3 | 13.4 | 0.37 | 0.37 | 3 | 296 |
| 60 | 0.2 | 26.1 | 0.19 | 0.19 | 3 | 405 |
| | E5n100 | | | | | |
| 1 | 0.8 | 0.1 | 55.17 | 55.17 | 5 | 1 |
| 2 | 0.5 | 0.2 | 24.13 | 24.11 | 4 | 4 |
| 5 | 0.4 | 0.7 | 6.93 | 6.91 | 3 | 28 |
| 10 | 0.4 | 1.2 | 2.54 | 2.54 | 4 | 42 |
| 20 | 0.3 | 4.2 | 0.90 | 0.90 | 3 | 125 |
| 30 | 0.4 | 8.1 | 0.47 | 0.47 | 4 | 203 |
| 40 | 0.4 | 12.9 | 0.29 | 0.29 | 4 | 288 |
| 60 | 0.2 | 29.0 | 0.13 | 0.13 | 3 | 442 |
| | E4n500 | | | | | |
| 1 | 6.0 | 2.2 | 194.07 | 194.05 | 3 | 1 |
| 2 | 11.4 | 6.9 | 135.18 | 133.20 | 4 | 32 |
| 5 | 11.5 | 5.2 | 76.01 | 75.79 | 4 | 19 |
| 10 | 10.1 | 5.1 | 44.34 | 44.33 | 4 | 15 |
| 20 | 9.0 | 12.8 | 21.67 | 21.68 | 4 | 39 |
| 30 | 8.9 | 27.1 | 13.02 | 13.03 | 4 | 68 |
| 40 | 9.1 | 47.5 | 8.80 | 8.81 | 4 | 103 |
| 60 | 11.3 | 133.9 | 4.94 | 4.95 | 5 | 182 |
| 100 | 7.1 | 429.0 | 2.33 | 2.34 | 4 | 360 |

(e) *Comparison with majorization method.* We use the following stopping criterion in `Major`:

$$(f^k)^2 \le \mathrm{tol}_{\mathrm{m}} \quad \text{or} \quad (f^{k-1})^2/(f^k)^2 - 1 \le \mathrm{tol}_{\mathrm{m}},$$

where $\mathrm{tol}_{\mathrm{m}} = 10^{-4}$. This criterion has been suggested by [33] for better performance of `Major`. The results over different $r$ and $n$ are listed in Tables 6 and 7. One observation is that the number of iterations in `SemiNewton` is quite small (less than 10 iterations). This may be justified on two fronts.

TABLE 7
*Comparison between* `SemiNewton` *and* `Major`.

| | $t$ | | $f$ | | iter | |
|---|---|---|---|---|---|---|
| $r$ | `SemiNewton` | `Major` | `SemiNewton` | `Major` | `SemiNewton` | `Major` |
| | E5n500 | | | | | |
| 1 | 33.2 | 2.2 | 384.16 | 384.16 | 8 | 1 |
| 2 | 18.0 | 2.6 | 205.16 | 204.76 | 5 | 4 |
| 5 | 11.7 | 4.4 | 68.98 | 68.78 | 4 | 14 |
| 10 | 11.4 | 7.5 | 26.28 | 26.26 | 4 | 27 |
| 20 | 9.7 | 15.1 | 9.54 | 9.54 | 4 | 47 |
| 30 | 6.9 | 33.0 | 5.22 | 5.23 | 3 | 84 |
| 40 | 10.9 | 59.0 | 3.40 | 3.40 | 5 | 129 |
| 60 | 8.8 | 168.3 | 1.85 | 1.86 | 4 | 230 |
| 100 | 6.5 | 934.8 | 0.45 | 0.46 | 4 | 803 |
| | E4n1000 | | | | | |
| 1 | 32.6 | 16.4 | 394.03 | 394.03 | 3 | 1 |
| 2 | 79.6 | 30.8 | 278.91 | 274.72 | 5 | 37 |
| 5 | 99.0 | 26.5 | 165.77 | 165.29 | 6 | 24 |
| 10 | 78.0 | 24.1 | 107.09 | 107.04 | 5 | 16 |
| 20 | 66.3 | 32.4 | 62.48 | 62.50 | 5 | 21 |
| 30 | 51.2 | 61.0 | 42.11 | 42.13 | 4 | 40 |
| 40 | 50.7 | 100.9 | 30.49 | 30.52 | 4 | 63 |
| 60 | 47.5 | 274.6 | 18.32 | 18.35 | 4 | 113 |
| 100 | 49.4 | 1141.8 | 9.04 | 9.08 | 4 | 229 |
| 200 | 48.6 | 8996.7 | 3.28 | 3.32 | 4 | 553 |
| | E5n1000 | | | | | |
| 1 | 153.8 | 16.8 | 714.13 | 714.13 | 8 | 1 |
| 2 | 84.4 | 17.6 | 356.31 | 355.96 | 5 | 4 |
| 5 | 65.2 | 22.4 | 109.13 | 109.02 | 4 | 14 |
| 10 | 54.2 | 28.8 | 39.77 | 39.78 | 4 | 25 |
| 20 | 38.9 | 68.0 | 14.13 | 14.13 | 3 | 66 |
| 30 | 39.5 | 151.4 | 7.70 | 7.71 | 3 | 121 |
| 40 | 39.8 | 263.7 | 5.01 | 5.02 | 3 | 184 |
| 60 | 47.5 | 755.0 | 2.73 | 2.74 | 4 | 323 |
| 100 | 45.2 | 3296.2 | 1.27 | 1.28 | 4 | 613 |
| 200 | 51.1 | 20141.2 | 0.44 | 0.46 | 4 | 1241 |

First, the augmented Lagrangian method for nonlinear semidefinite programming is potentially superlinearly convergent, provided that the constraint nondegeneracy holds among other conditions. For a general theory on its convergence rate for nonlinear SDPs, see Sun, Sun, and Zhang [35]. One such result has been detailed in [27, Thm. 3.4] for convex SDPs. However, for our case it is difficult to verify whether the augmented Lagrangian is superlinearly convergent. Second, subproblem (22) can be solved successfully by the semismooth Newton method, giving a meaningful $X^{k+1}$. This leads to a very effective update of $U^{k+1}$, which in turn speeds up the progress of our algorithm. Moreover, the algorithm begins with a very good starting point. Compared with `Major`, the function values given by two methods are more or less the same; therefore the main concern is focused on the CPUtime. From Table 6, one can find that for smaller $n$ and $r$, `Major` is very fast. However, as $r$ increases, the CPUtime for `Major` grows at a dramatic rate. In contrast, `SemiNewton` takes about the same amount of time as $r$ becomes bigger. This trend is more obvious when the size of the problem gets bigger, as shown in Table 7. For example, in the case of E5n1000 with $r = 20$, `SemiNewton` has better performance in the sense that it takes about 40s for `SemiNewton`

TABLE 8
*Comparison between* `SemiNewton` *and* `CMajor`*: E6*$n$100.

| | $t$ | | $f$ | | iter | |
|---|---|---|---|---|---|---|
| $r$ | `SemiNewton` | `CMajor` | `SemiNewton` | `CMajor` | `SemiNewton` | `CMajor` |
| 18 | 1.0 | 9.0 | 19.32 | 20.27 | 4 | 10 |
| 25 | 0.9 | 22.6 | 14.35 | 14.81 | 4 | 23 |
| 40 | 0.9 | 20.6 | 8.41 | 8.71 | 5 | 18 |
| 60 | 0.5 | 81.7 | 4.07 | 4.23 | 4 | 30 |
| 80 | 0.4 | 113.8 | 1.21 | 1.27 | 4 | 35 |

to return a solution of the same function value, compared to 68s by `Major`. When $r = 30$, `SemiNewton` used about 40s, while `Major` used about 151s.

(f) *Comparison with majorization method in* [33] *for more equality constraints.* We also note that Algorithm 4.2 is readily extended to include extra equality constraints, implying that Algorithm 3.1 is capable of dealing with problem (1) with more equality constraints. Such problems are considered in [33], where the authors extended majorization method (referred to as `CMajor` here) to solve it. We tested `CMajor` and `SemiNewton` on E6, which is taken from [33]. The results in Table 8 show a significant improvement of `SemiNewton` over `CMajor`, especially in CPUtime. It is worth pointing out that `CMajor` is restricted to equality constraints of the very special case $X_{ij} = 0$, whereas `SemiNewton` is capable of handling general linear equality constraints. For the latter case, the corresponding version of subproblem (20) will have to be solved by other methods (e.g., the smoothing Newton method of [10]) rather than Algorithm 4.2.

**6. Conclusion.** In this paper, a novel approach is introduced to solve the nearest low-rank correlation matrix problem (1). The resulting sequential semismooth Newton method guarantees to produce a stationary point of the nonlinear semidefinite programming problem (4), which is an equivalent reformulation of (1). Our numerical results show that the method is highly efficient and outperforms the best available methods for (1). The key to the success of our method is the equivalent reformulation of problem (1) as well as the simple update of $U$ and the fast semismooth Newton method used to solve the subproblem at each iteration. The semismooth Newton method depends heavily on the subproblem being reformulated as a least-square problem.

This elicits our first question, How can our method be extended to the nearest low-rank correlation matrix problem with $H$-weighting,

$$\begin{array}{ll} \min\limits_{X\in\mathcal{S}^n} & \frac{1}{2}\|H\circ(X-C)\|^2 \\ \text{s.t.} & \text{diag}(X) = e,\ X \succeq 0, \\ & \text{rank}(X) \le r? \end{array}$$

For the nearest correlation matrix problem (3), this has been done in [27]. For the extension, one has to deal with the $H$-weighted version of subproblem (20). The constraints remain the same, but we have the $H$-weighted objective function, denoted by

$$L^H_{(U,\mu,c)}(X) := \frac{1}{2}\|H\circ(X-C)\|^2 - \mu\Big(\langle X,U\rangle - n\Big) + \frac{c}{2}\Big(\langle X,U\rangle - n\Big)^2.$$

The dual (see subsection 4.1) of this $H$-weighted subproblem is not unconstrained anymore due to the fact that the orthogonal projection onto the positive semidefinite cone

under the $H$-weighting does not usually have a closed-form formula. Consequently, the semismooth Newton method developed in sub tion 4.2 is not appropriate for the weighted subproblem. One has to explore some other options. For example, Gao and Sun [11] proposed a penalized majorization method for the $H$-weighted problem.

The second question is how to characterize the local and global minima of the new reformulation (4). Such characterizations will lead to significant improvement in moving the obtained stationary point toward a local/global minimum of (4). We feel this is a difficult question but is worth serious investigation.

**Acknowledgments.** We would like to thank Prof. Defeng Sun of National University of Singapore for bringing up the idea to treat the subproblem (20) as a least-square problem so that the semismooth Newton method could be developed. We also would like to thank Dr. Zhaosong Lu of Simon Fraser University for bringing [20] to our attention and for the helpful discussion on the feasibility issue in augmented Lagrangian methods. We also thank the two referees for their valuable comments and constructive suggestions, which have significantly improved the quality of the paper.

## REFERENCES

[1] F. Alizadeh, J.-P.A. Haeberly, and M.L. Overton, *Complementarity and nondegeneracy in semidefinite programming*, Math. Programming, 77 (1997), pp. 111–128.
[2] J.F. Bonnans and A. Shapiro, *Perturbation Analysis of Optimization Problems*, Springer, New York, 2000.
[3] R. Borsdorf and N.J. Higham, *A preconditioned Newton algorithm for the nearest correlation matrix*, IMA J. Numer. Anal., 30 (2010), pp. 94–107.
[4] S. Boyd and L. Xiao, *Least-squares covariance matrix adjustment*, SIAM J. Matrix Anal. Appl., 27 (2005), pp. 532–546.
[5] D. Brigo, *A Note on Correlation and Rank Reduction*, http://www.damianobrigo.it (2002).
[6] D. Brigo and F. Mercurio, *Interest Rate Models: Theory and Practice*, Springer-Verlag, Berlin, 2001.
[7] F.H. Clarke, *Optimization and Nonsmooth Analysis*, John Wiley & Sons, New York, 1983.
[8] B.C. Eaves, *On the basic theorem for complementarity*, Math. Programming, 1 (1971), pp. 68–75.
[9] B. Flury, *Common Principle Components and Related Multivariate Models*, John Wiley & Sons, New York, 1988.
[10] Y. Gao and D. Sun, *Calibrating least squares semidefinite programming with equality and inequality constraints*, SIAM J. Matrix Anal. Appl., 31 (2009), pp. 1432–1457.
[11] Y. Gao and D.F. Sun, *A Majorized Penalty Approach for Calibrating Rank Constrained Correlation Matrix Problems*, Technical report, Department of Mathematics, National University of Singapore, Singapore, 2010.
[12] I. Grubišić and R. Pietersz, *Efficient rank reduction of correlation matrices*, Linear Algebra Appl., 422 (2007), pp. 629–653.
[13] D. Henrion and J. Malick, *Projection methods in conic optimization*, in Handbook of Semidefinite, Cone and Polynomial Optimization, M. Anjos and J. Lasserre, eds., 2012, Springer, pp. 565–600.
[14] M.R. Hestenes and E. Stiefel, *Methods of conjugate gradients for solving linear systems*, J. Research Nat. Bur. Standards, 49 (1952), pp. 409–436.
[15] N.J. Higham, *Computing the nearest correlation matrix—A problem from finance*, IMA J. Numer. Anal., 22 (2002), pp. 329–343.
[16] J.-B. Hiriart-Urruty and D. Ye, *Sensitivity analysis of all eigenvalues of a symmetric matrix*, Numer. Math., 70 (1995), pp. 45–72.
[17] J.C. Hull and A. White, *Forward rate volatilities, swap rate volatilities, and implementation of the LIBOR market model*, J. Fixed Income, 10 (2000), pp. 46–62.
[18] C.T. Kelley, *Solving Nonlinear Equations with Newton's Method*, Fundam. Algorithms 1, SIAM, Philadelphia, 2003.
[19] B. Kummer, *Newton's method for nondifferentiable functions*, in Advances in Mathematical Optimization, Math. Res. 45, Akademie-Verlag, Berlin, 1988, pp. 114–125.

[20] Z. Lu and Y. Zhang, *An augmented Lagrangian approach for sparse principal component analysis*, Math. Programm., to appear.
[21] J. Malick, *A dual approach to semidefinite least-squares problems*, SIAM J. Matrix Anal. Appl., 26 (2004), pp. 272–284.
[22] J. Nocedal and S.J. Wright, *Numerical Optimization*, Springer, New York, 1999.
[23] M. L. Overton and R.S. Womersley, *On the sum of the largest eigenvalues of a symmetric matrix*, SIAM J. Matrix Anal. Appl., 13 (1992), pp. 41–45.
[24] M. Overton and R.S. Womersley, *Optimality conditions and duality theory for minimizing sums of the largest eigenvalues of symmetric matrices*, Math. Programming, 62 (1993), pp. 321–357.
[25] R. Pietersz and P.J.F. Groenen, *Rank reduction of correlation matrices by majorization*, Quant. Finance, 4 (2004), pp. 649–662.
[26] H. Qi and D. Sun, *A quadratically convergent Newton method for computing the nearest correlation matrix*, SIAM J. Matrix Anal. Appl., 28 (2006), pp. 360–385.
[27] H.D. Qi and D.F. Sun *An augmented Lagrangian dual approach for the H-weighted nearest correlation matrix problem*, IMA J. Numer. Anal., 31 (2011), pp. 491-511.
[28] L. Qi and J. Sun, *A nonsmooth version of Newton's method*, Math. Programming, 58 (1993), pp. 353–367.
[29] F. Rapisarda, D. Brigo, and F. Mercurio, *Parameterizing correlations: A geometric interpretation*, IMA J. Manag. Math., 18 (2007), pp. 55–73.
[30] R. Rebonato, *On the simultaneous calibration of multifactor lognormal interest rate models to Black volatilities and to the correlation matrix*, J. Comput. Finance, 2 (1999), pp. 5–27.
[31] R. Rebonato, *Modern Pricing and Interest-Rate Derivatives*, Princeton University Press, Princeton, NJ, 2002.
[32] R. Rebonato, *Interest-rate term-structure pricing models: A review*, Proc. Roy. Soc. Lond. Ser. A Math. Phys. Eng. Sci., 460 (2004), pp. 667–728.
[33] D. Simon and J.A. Abell, *A majorization algorithm for constrained correlation matrix approximation*, Linear Algebra Appl., 432 (2010), pp. 1152–1164.
[34] D.F. Sun and J. Sun, *Semismooth matrix valued functions*, Math. Oper. Res., 27 (2002), pp. 150–169.
[35] D.F. Sun, J. Sun, and L.-W. Zhang, *The rate of convergence of augmented Lagrangian method in nonlinear semidefinite programming*, Math. Program., 114 (2008), pp. 349–391.
[36] K.C. Toh, *An inexact path-following algorithm for convex quadratic SDP*, Math. Program., 112 (2008), pp. 221–254.
[37] K.C. Toh, R.H. Tütüncü, and M.J. Todd, *Inexact primal-dual path-following algorithms for a special class of convex quadratic SDP and related problems*, Pac. J. Optim., 3 (2007), pp. 135–164.
[38] L. Wu, *Fast at-the-money calibration of the LIBOR market model using Lagrangean multipliers*, J. Comput. Finance, 6 (2002/3), pp. 39–77.
[39] Z. Zhang and L. Wu, *Optimal low-rank approximation to a correlation matrix*, Linear Algebra Appl., 364 (2003), pp. 161–187.
[40] X.-Y. Zhao, D. Sun, and K.-C. Toh, *A Newton-CG augmented Lagrangian method for semidefinite programming*, SIAM J. Optim., 20 (2010), pp. 1737–1765.