

A CONVEX MATRIX OPTIMIZATION FOR THE ADDITIVE CONSTANT PROBLEM IN MULTIDIMENSIONAL SCALING WITH APPLICATION TO LLE*

HOU-DUO QI†

Abstract. The additive constant problem has a long history in multi-dimensional scaling and it has recently been used to resolve the issue of indefiniteness of the geodesic distance matrix in ISOMAP. But it would lead to a large positive constant being added to all eigenvalues of the centered geodesic distance matrix, often causing significant distortion of the original distances. In this paper, we reformulate the problem as a convex optimization of almost negative semidefinite matrix so as to achieve minimal variation of the original distances. We then develop a Newton-CG method and further prove its quadratic convergence. Finally, we include a novel application to the famous LLE (Locally Linear Embedding in nonlinear dimensionality reduction), addressing the issue when the input of LLE has missing values. We justify the use of the developed method to tackle this issue by establishing that the local Gram matrix used in LLE can be obtained through a local Euclidean distance matrix. The effectiveness of our method is demonstrated by numerical experiments

Key words. Euclidean distance matrix, semismooth Newton method, quadratic convergence, dimensionality reduction, LLE.

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

1. Introduction. It is probably not too exaggerating to say that ISOMAP [34] and LLE [26] – two most famous methods in nonlinear dimensionality reduction, have helped extensive use of classical Multidimensional Scaling (cMDS) in machine learning. In particular, ISOMAP uses the geodesic distance to approximate the pairwise Euclidean distance on a manifold and then applies cMDS to the geodesic distance matrix. If the matrix is close to be Euclidean, ISOMAP would work perfectly fine. Otherwise, a certain modification is necessary. One of the popular proposals is rooted in the additive constant problem (ACP) in MDS [35]. However, it often leads to a large positive number being added to all the eigenvalues of the centered geodesic distance matrix (see, e.g., [9, 30] and [36, Sect. 8.1.3]), causing significant distortion of the original distances. As for LLE with its inputs being pairwise distances/dissimilarities [27, Sect. 5.1], the neighbouring distance matrix at each point is assumed to be Euclidean. However, those distance matrices are rarely Euclidean in practice, say due to missing values. Once again, certain modification is needed. The purpose of this paper is to reformulate a variant of ACP as a convex matrix optimization, which results in a smaller constant to be added and hence causes less distortion. We will develop a fast method for this matrix optimization problem and demonstrate its application to LLE. Below we first give a brief introduction of ACP and its variants.

1.1. ACP and cMDS. We start with cMDS. Suppose there are n points \mathbf{x}_i in \mathbb{R}^r . Let D be the matrix of squared Euclidean distances: $D_{ij} := d_{ij}^2 = \|\mathbf{x}_i - \mathbf{x}_j\|^2$. Such matrices are called Euclidean. The theory of MDS implies that one can actually generate a set of such points \mathbf{x}_i from D by decomposing the centralizing matrix $B := -\frac{1}{2}JDJ$, where $J := I - \frac{1}{n}ee^T$ (“=” and “:=” mean “define”) is the centering matrix and e is the column vector of all ones in \mathbb{R}^n . In fact, it follows from [28, 37] that the matrix B is positive semidefinite and a set of embedding points \mathbf{x}_i can be

*This version: October 7, 2015.

†School of Mathematics, University of Southampton, Highfield, Southampton SO17 1BJ, UK.
E-mail: hdqi@soton.ac.uk.

obtained via

$$B = P \text{diag}(\lambda_1, \dots, \lambda_n) P^T =: X^T X \quad \text{with } X := [\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_n],$$

where $\lambda_1 \geq \dots \geq \lambda_n \geq 0$ are the eigenvalues of B , $P^T P = I$, and the i th column of X is \mathbf{x}_i . When D is not Euclidean, cMDS simply orthogonally projects B to the positive semidefinite cone \mathcal{S}_+^n in the $n \times n$ symmetric matrix space \mathcal{S}^n endowed with the standard inner product. The projection is denoted by $\Pi_{\mathcal{S}_+^n}(B)$ and is given by

$$\Pi_{\mathcal{S}_+^n}(B) := P \text{diag}(\lambda_1^+, \dots, \lambda_n^+) P =: X^T X, \quad (1)$$

where $\lambda_i^+ := \max\{\lambda_i, 0\}$. The famous ISOMAP can be cast as the direct application of (1) with d_{ij} being given by the geodesic distance from point i to point j on a manifold.

The indefiniteness of B may be from various reasons. Torgerson [35] argues that all the distances d_{ij} should be simultaneously added by a constant to become $\hat{d}_{ij} := d_{ij} + c$ so that $\hat{D} := (\hat{d}_{ij}^2)$ becomes Euclidean. The principle is that the smaller c is, the better. This is the origin of ACP. The question of existence as well as how to compute such c was only settled 30 years later by Cailliez [6], who proved that the smallest c is the largest eigenvalue of $2n \times 2n$ matrix

$$\hat{B} := \begin{bmatrix} 0 & 2B \\ -I & -2J\sqrt{D}J \end{bmatrix},$$

where \sqrt{D} is the componentwise square root of D (note: $D_{ij} \geq 0$). This result was first used by Choi and Choi [9] to tackle the indefiniteness of ISOMAP.

There were also other developments on ACP. Lingo [19] proposed a slightly different version. He argues that the constant can be added to d_{ij}^2 and it would result in a simpler solution: $(-c)$ is the smallest eigenvalue of B (note: the smallest eigenvalue must be negative; otherwise D would be Euclidean already). In both the computation of [19, 6], the distance d_{ij} remains fixed. Cooper [11] numerically demonstrated that it is more favourable if it also allows d_{ij} be varied, resulting in a non-convex optimization problem. There are advantages and disadvantages of those methods and we do not intend to discuss them in detail here. However, we would like to point out one major weakness in those methods in that they tend to yield large corrections even there were only a small number of distances that have large deviations from their true distances (they are known as outliers). This phenomenon is well demonstrated in the following example.

We generate a small network of 15 points. The first point is at the origin and the remaining 14 points are evenly generated on the unit circle. Suppose there is just one distance from the origin that has a large deviation. For example, let $d(1, 15) = 4$ (the true distance is 1). Figure 1.1 shows the reconstructions of the network by the 4 methods: cMDS (1), Lingo [19], Cailliez [6], and the method LLEMDS to be proposed in this paper. It is obvious that only LLEMDS correctly recovered the underlying structure. The major reason for the observed distortion from the other methods is that the correction constant c is too big with $c = 12.5812$ by Lingo [19] and $c = 6.1234$ by Cailliez [6]. Our method yielded a smaller constant $c = 1.2071$. This example also justifies the routine used in [9, 30] that an outlier removal procedure is necessary before the Cailliez method can be used to ISOMAP. We now explain our proposal.

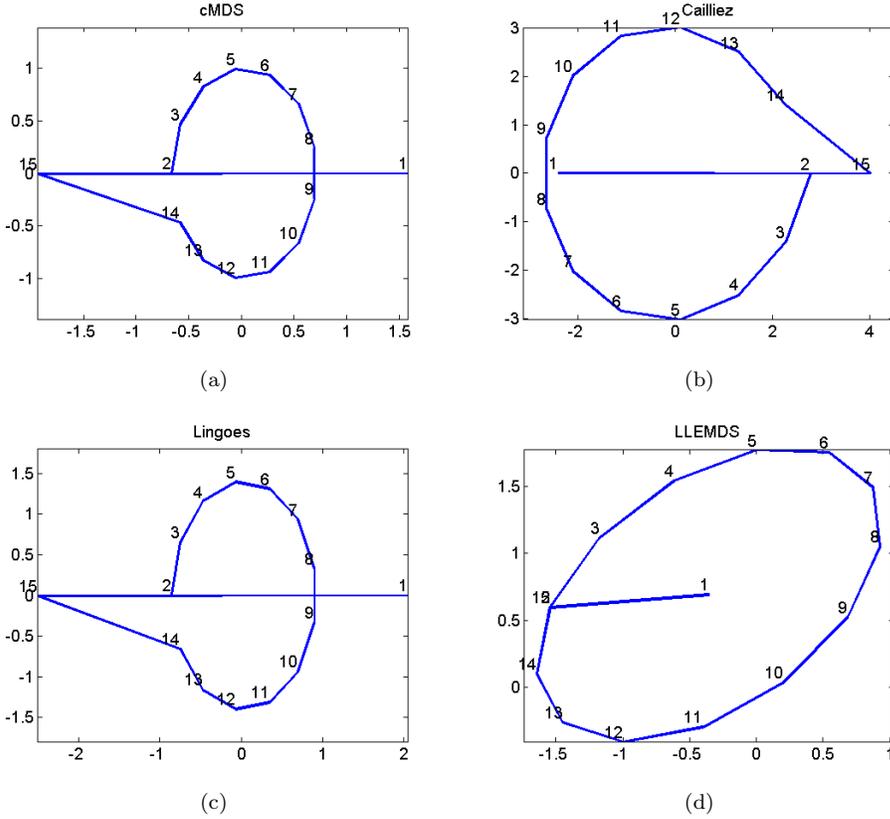


FIG. 1.1. Reconstruction of a small network by the 4 methods. Fig. 1.1a is by cMDS in (1); Fig. 1.1b is by the method of Cailliez [6] with $c = 6.1234$; Fig. 1.1c is by the method of Lingoes [19] with $c = 12.5812$. Each of those methods has resulted in a certain level of distortion of the original network. Fig. 1.1d by our method with $c = 1.2071$ correctly reveals the original network structure.

1.2. Our formulation and relevant research. Suppose $\widehat{D} = (\widehat{d}_{ij}^2)$, $i, j = 1, \dots, n$ contains the observed squared pairwise distances. The purpose is to find a true Euclidean distance matrix $D = (d_{ij}^2)$ and a constant c such that (throughout the paper, \widehat{D} is reserved for the observed distance matrix)

$$d_{ij}^2 \approx \widehat{d}_{ij}^2 + c \quad \forall i \neq j. \quad (2)$$

This approximation that allows changes both in d_{ij} and c follows the line of argument of Cooper [11], where d_{ij}^2 was represented by $\|\mathbf{x}_i - \mathbf{x}_j\|^2$ with $\mathbf{x}_i \in \mathbb{R}^r$. When using the least-square principle to (2), we obtain

$$\min \frac{1}{2} \sum_{i \neq j} \left(D_{ij} - (\widehat{D}_{ij} + c) \right)^2, \quad \text{s.t. } D \text{ is Euclidean, } c \in \mathbb{R}, \quad (3)$$

which does not include the diagonal part ($i = j$). Unfortunately, this least-square reformulation does not lead to any new improvement because it always has zero as its optimal objective value and the Lingoes solution [19] would be one of the optimal solutions (see the proof of Prop. 2.1). In order to achieve a good improvement, we

include the diagonal part below:

$$\min \frac{1}{2} \sum_{i \neq j} \left(D_{ij} - (\widehat{D}_{ij} + c) \right)^2 + \frac{1}{2} n c^2, \quad \text{s.t. } D \text{ is Euclidean, } c \in \mathbb{R}. \quad (4)$$

We will show that the optimal constant thus obtained must be less than the Lingoes constant [19]. Below, we explain how to reformulate (4) into a form that can be efficiently solved.

The greatest advantage in using the squared distances is that the Euclidean matrix D has a nice characterization due to Schoenberg [28, 29] and Young and Householder [37]:

$$D \text{ is Euclidean} \iff D \in \mathcal{K}_-^n \text{ and } \text{diag}(D) = 0, \quad (5)$$

where $\text{diag}(D)$ is the diagonal vector of D and \mathcal{K}_-^n is the *almost negative semidefinite cone* defined by (see also [20, Def. 2.2])

$$\mathcal{K}_-^n := \{A \in \mathcal{S}^n \mid \mathbf{v}^T A \mathbf{v} \leq 0 \ \forall \mathbf{v} \in e^\perp\} = \{A \in \mathcal{S}^n \mid -JAJ \in \mathcal{S}_+^n\} \quad (6)$$

with $e^\perp \subset \mathbb{R}^n$ being the subspace orthogonal to the vector e . The second characterization in (6) is due to J being the projection matrix to e^\perp . Understandably, the cone $\mathcal{K}_+^n := -\mathcal{K}_-^n$ is called the almost positive semidefinite cone.

It follows from (5) and the fact $Je = 0$ that

$$\{A - cee^T \mid A \text{ is Euclidean and } c \in \mathbb{R}\} = \{Y \mid Y \in \mathcal{K}_-^n \text{ and } Y_{11} = \dots = Y_{nn}\}. \quad (7)$$

Consequently, problem (4) is equivalent to

$$\min \frac{1}{2} \|Y - \widehat{D}\|^2, \quad \text{s.t. } Y \in \mathcal{K}_-^n \text{ and } Y_{11} = \dots = Y_{nn}, \quad (8)$$

where the norm is the Frobenius norm. The objective in (8) is due to the observation:

$$\sum_{i \neq j} \left(D_{ij} - (\widehat{D}_{ij} + c) \right)^2 + n c^2 = \sum_{i,j} \left((D_{ij} - c) - \widehat{D}_{ij} \right)^2 = \|(D - cee^T) - \widehat{D}\|^2$$

using the fact $\text{diag}(\widehat{D}) = \text{diag}(D) = 0$. The feasible region of (8) is the cone of the almost negative semidefinite matrices with constant diagonals. The relationship between the solutions of (4) and (8) is $D = Y + cee^T$ with $c = -Y_{11}$.

Problem (8) is the core problem that we are going to solve in this paper and it can be cast as the orthogonal projection of a given point \widehat{D} onto a closed convex cone intersected with a hyperplane. Such problems can often be efficiently solved by a Newton-CG method as previously demonstrated in [23, 38, 21], where problems of similar structures have been studied. A key requirement is that the matrix in defining the Newton equation should be positive definite, guaranteeing the quadratic convergence of Newton's method. We will show that it is the case for Problem (8).

1.3. Application to LLE and organization of the paper. The proposed method has several direct applications. For example, it can be used to tackle the indefiniteness issue in ISOMAP. However, we would like to present a novel application to LLE [26, 27] on an issue that has not been well addressed in literature. Usually, the input of LLE are the high dimensional data matrix $X = [\mathbf{x}_1, \dots, \mathbf{x}_n]$ with the

coordinates $\mathbf{x}_i \in \mathbb{R}^N$ being known. We denote it by $\text{LLE}(X)$. However, in practice, the data matrix such as the genetic data studied in [31, 5] often has missing values. A common approach is to impute all the missing values. When the missing values appear across a large portion of the dimensions, a set of low dimensions would be left after imputation, rendering the remaining data invalid for use.

In this paper, we propose a new approach in handling the missing values in X . We first calculate the squared pairwise distances using the data available in X . This would generate the distance matrix \widehat{D} . $\text{LLE}(\widehat{D})$, described in [27, Sect. 5.1], assumes that \widehat{D} is Euclidean. We will prove that when X has no missing values, $\text{LLE}(X)$ and $\text{LLE}(\widehat{D})$ actually produce a same set of embedding points. This result immediately suggests to make \widehat{D} Euclidean if it is not yet so. Furthermore, what is needed in $\text{LLE}(\widehat{D})$ is that the local distance matrix \widehat{D}_i at each embedding point should be Euclidean. Therefore, we can apply the proposed method on \widehat{D}_i to make it Euclidean. The advantage on working on \widehat{D}_i is that the size of \widehat{D}_i is usually small and is decided by the neighborhood size chosen in LLE. A disadvantage is that there are as many as n such \widehat{D}_i 's. Therefore, the speed of algorithms for (8) is crucial to this application. Fortunately, the fast Newton-CG method proposed in this paper is up to this challenge. We will develop and numerically demonstrate this application in details.

The paper is organized as follows. In the next section, we present some basic results on the almost negative semidefinite cone \mathcal{K}_-^n , especially on the generalized Jacobian of the projection function on it. In Sect. 3, we study the constraint nondegeneracy, a key property that ensures the quadratic convergence of the Newton-CG method, which is described in Sect. 4. The promised novel application to LLE in developed in Sect. 5 with numerical demonstrations. We conclude the paper in Sect. 6.

2. Preliminaries and Some Basic Results on \mathcal{K}_-^n . We first list some notation that we will use in this paper. We use I_n (to indicate the dimension involved whenever it is necessary) or I to denote the identity matrix in \mathcal{S}^n . For a matrix $A \in \mathcal{S}^n$, we let $\text{Tr}(A)$ be the trace of A . Let $\mathcal{S}_-^n := -\mathcal{S}_+^n$ and the vectors be denoted by bold faced lower letters and treated as column vectors. For example, \mathbf{x}^T will be a row vector. $\text{Diag}(\mathbf{x})$ is the diagonal matrix formed by \mathbf{x} . $A \circ B := [A_{ij}B_{ij}]$ is the Hadamard product between two matrices A and B of same size. For subsets α, β of $\{1, \dots, n\}$, denote $A_{\alpha\beta}$ as the submatrix of A indexed by α and β (α for rows and β for columns). $|\alpha|$ is the cardinality of α .

There are two important matrices that will play an important role in our analysis. One is the centralizing matrix J used to define the matrix B in (1). Another is the Householder transformation Q that maps the vector $e \in \mathbb{R}^n$ to the vector $[0, \dots, 0, -\sqrt{n}]^T \in \mathbb{R}^n$. Let $\mathbf{v} := [1, \dots, 1, 1 + \sqrt{n}]^T \in \mathbb{R}^n$. Then

$$Q = I - \frac{2}{\mathbf{v}^T \mathbf{v}} \mathbf{v} \mathbf{v}^T. \quad (9)$$

The matrices J and Q have the following properties:

$$J^2 = J, \quad J e = 0, \quad Q^2 = I \quad \text{and} \quad J = Q \begin{bmatrix} I_{n-1} & 0 \\ 0 & 0 \end{bmatrix} Q. \quad (10)$$

The orthogonal projection $\Pi_{\mathcal{K}_-^n}(X)$ of X onto the almost negative semidefinite cone \mathcal{K}_-^n also plays a vital role in this paper and is defined as follows:

$$\Pi_{\mathcal{K}_-^n}(X) := \arg \min_Y \|X - Y\| \quad \text{s.t.} \quad Y \in \mathcal{K}_-^n.$$

A nice property is that this projection can be done through the orthogonal projection $\Pi_{\mathcal{S}_+^n}(\cdot)$ onto the positive semidefinite cone \mathcal{S}_+^n and it is due to Gaffke and Mathar [14]

$$\Pi_{\mathcal{K}_-^n}(X) = -\Pi_{\mathcal{K}_+^n}(-X) = X - \Pi_{\mathcal{S}_+^n}(JXJ) \quad \forall X \in \mathcal{S}^n, \quad (11)$$

which with [24, Lemma 2.1(i)] implies that

$$-J\Pi_{\mathcal{K}_-^n}(X)J = J\Pi_{\mathcal{K}_+^n}(-X)J = \Pi_{\mathcal{S}_+^n}(-JXJ). \quad (12)$$

The remaining of the section is devoted to three tasks. The first is about our optimization model (4) showing that it yields a smaller constant than the Lingoes constant [19]. The second task is on the description of the generalized Jacobian of $\Pi_{\mathcal{K}_-^n}(\cdot)$ when cast as a function. The third one is on the description of the tangent cone of \mathcal{K}_-^n . Both descriptions are such formulated that they will be convenient for our quadratic convergence analysis of Newton's method later on.

2.1. On Model (4). First of all, we note that Problem (4) has a unique solution. This is because its equivalent problem (8) is strongly convex and there is one-to-one correspondence between their solutions. We now report a result that shows Problem (4) yields a smaller constant than the Lingoes constant [19].

PROPOSITION 2.1. *Let c_L and \bar{c} be the positive constants obtained respectively by the Lingoes method [19] and the Cailliez method [6]. Let (D_N, c_N) be the optimal solution of (4). We must have*

$$c_N^2 \leq c_L^2 \quad \text{and} \quad c_N^2 \leq \frac{1}{n} \sum_{i \neq j} \hat{d}_{ij}^2 (1 - \hat{d}_{ij})^2 + \bar{c}^2.$$

Proof. According to Lingoes [19] that there exists a constant $c_L > 0$ such that

$$D_L := \hat{D} + c_L ee^T - c_L I$$

is Euclidean. Hence, (D_L, c_L) is feasible with respect to Problem (4). It is also an optimal solution to (3) as it would achieve the optimal objective value *zero*. The optimality of (D_N, c_N) yields

$$\begin{aligned} nc_N^2 &\leq nc_N^2 + \sum_{i \neq j} ((D_N)_{ij}^2 - (\hat{D}_{ij} + c_N))^2 \\ &\leq nc_L^2 + \sum_{i \neq j} ((D_L)_{ij}^2 - (\hat{D}_{ij} + c_N))^2 \\ &= nc_L^2. \end{aligned}$$

Now we prove the second part of the claim. According to Calilliez [6] that the matrix

$$\bar{D} := \left(\sqrt{\hat{D}} + \bar{c} ee^T - \bar{c} I \right) \circ \left(\sqrt{\hat{D}} + \bar{c} ee^T - \bar{c} I \right)$$

is Euclidean. It follows from [29, Cor. 1] (see also [22, page 166 (R3)]) that the componentwise square root matrix $\sqrt{\bar{D}}$ is also Euclidean. Therefore, $(\sqrt{\bar{D}}, \bar{c})$ is feasible with respect to Problem (4). The optimality of (D_N, c_N) yields

$$\begin{aligned} nc_N^2 &\leq nc_N^2 + \sum_{i \neq j} ((D_N)_{ij}^2 - (\hat{D}_{ij} + c_N))^2 \leq n\bar{c}^2 + \sum_{i \neq j} ((\sqrt{\bar{D}})_{ij} - (\hat{D}_{ij} + \bar{c}))^2 \\ &= n\bar{c}^2 + \sum_{i \neq j} (\sqrt{\hat{D}_{ij}} + \bar{c} - (\hat{D}_{ij} + \bar{c}))^2 = n\bar{c}^2 + \sum_{i \neq j} (\hat{d}_{ij} - \hat{d}_{ij}^2)^2. \end{aligned}$$

Dividing n on both sides of the inequalities leads to our claim in the proposition. \square

We note that Cailliez's constant \bar{c} is added to the distance \hat{d}_{ij} , while our constant c_N is added to the squared distance d_{ij}^2 . Our result shows that c_N is bounded by \bar{c} plus a positive term. It is a very rough bound as in our proof, we dropped the positive quantity $\sum_{i \neq j} ((D_N)_{ij}^2 - (\hat{D}_{ij} + c_N))^2$. This term may potentially force c_N much smaller than \bar{c} . This possibility has been verified in Fig. 1.1b. The constant c_N is also much smaller than c_L in Fig. 1.1c.

2.2. Generalized Jacobian of $\Pi_{\mathcal{K}^n}(\cdot)$. The projection operator $\Phi(X) := \Pi_{\mathcal{K}^n}(X)$ is Lipschitzian with constant 1 (i.e., nonexpensive). It is hence almost everywhere differentiable. The generalized Jacobian at any given point X , denoted by $\partial\Phi(X)$, is then defined to be the convex hull of all limiting Jacobians of the sequence $\{\Phi'(X^j)\}$, where X^j converges to X :

$$\partial\Phi(X) := \text{conv} \left\{ \lim_{X^j \rightarrow X} \Phi'(X^j) \mid \begin{array}{l} \text{for all sence } \{X^j\} \text{ such that} \\ \Phi'(X^j) \text{ exists and the limits exist} \end{array} \right\}.$$

Because of the formula (11), $\partial\Phi(X)$ can be obtained through the Jacobian of $\Pi_{\mathcal{S}_+^n}(\cdot)$ at (JXJ) . Fortunately, the structure of $\partial\Pi_{\mathcal{S}_+^n}(\cdot)$ has been well understood due to Sun [32, Prop. 2.2]. In order to apply this result, we need to handle the linear mapping JXJ . It follows from (10) that

$$JXJ = Q \begin{bmatrix} I_{n-1} & 0 \\ 0 & 0 \end{bmatrix} QXQ \begin{bmatrix} I_{n-1} & 0 \\ 0 & 0 \end{bmatrix} Q = Q \begin{bmatrix} \underline{X}_1 & 0 \\ 0 & 0 \end{bmatrix} Q, \quad (13)$$

where $\underline{X}_1 \in \mathcal{S}^{n-1}$ is the leading $(n-1) \times (n-1)$ block of the partition of QXQ :

$$\begin{bmatrix} \underline{X}_1 & \underline{\mathbf{x}} \\ \underline{\mathbf{x}}^T & \underline{x}_0 \end{bmatrix} := QXQ. \quad (14)$$

Let \underline{X}_1 have the following spectral decomposition:

$$\underline{X}_1 = W \text{Diag}(\lambda_1, \dots, \lambda_{n-1}) W^T, \quad (15)$$

where $\lambda_1 \geq \dots \geq \lambda_{n-1}$ are the eigenvalues of \underline{X}_1 and $W^T W = I_{n-1}$. Define the $n \times n$ orthogonal matrix \bar{W} by

$$\bar{W} := \begin{bmatrix} W & 0 \\ 0 & 1 \end{bmatrix}. \quad (16)$$

Then JXJ admits the following spectral decomposition:

$$JXJ = Q \bar{W} \text{Diag}(\lambda_1, \dots, \lambda_{n-1}, 0) \bar{W}^T Q. \quad (17)$$

Define three index sets which partition the $(n-1)$ eigenvalues $\{\lambda_i\}$:

$$\alpha := \{i : \lambda_i > 0\}, \quad \beta := \{i : \lambda_i = 0\} \quad \text{and} \quad \gamma := \{i : \lambda_i < 0\}. \quad (18)$$

For those eigenvalues, define the corresponding symmetric matrix $\Omega \in \mathcal{S}^{n-1}$ with entries

$$\Omega_{ij} := \frac{\max\{\lambda_i, 0\} + \max\{\lambda_j, 0\}}{|\lambda_i| + |\lambda_j|}, \quad i, j = 1, \dots, n \quad (19)$$

where $0/0$ is defined to be 1. Applying [32, Prop. 2.2] to the formula (11), we obtain a description for $\partial\Phi(X) = \partial\Pi_{\mathcal{K}_-^n}(X)$ below.

PROPOSITION 2.2. *For every element $V \in \partial\Phi(X)$, there exists a generalized Jacobian $\tilde{V} \in \partial\Pi_{\mathcal{S}_+^{|\beta|+1}}(0)$ such that*

$$VH = H - P\mathcal{W}_H P^T, \quad \forall H \in \mathcal{S}^n, \quad (20)$$

where $P := Q\overline{W}$, and

$$\mathcal{W}_H := \begin{bmatrix} W_\alpha^T \underline{H}_1 W_\alpha & [W_\alpha^T \underline{H}_1 W_\beta & 0] & \Omega_{\alpha\gamma} \circ W_\alpha^T \underline{H}_1 W_\gamma \\ \left[\begin{array}{c} W_\beta^T \underline{H}_1 W_\alpha \\ 0 \end{array} \right] & \tilde{V} \left(\left[\begin{array}{cc} W_\beta^T \underline{H}_1 W_\beta & 0 \\ 0 & 0 \end{array} \right] \right) & 0 \\ \Omega_{\alpha\gamma}^T \circ W_\gamma^T \underline{H}_1 W_\alpha & 0 & 0 \end{bmatrix}$$

and $\underline{H}_1 \in \mathcal{S}^{n-1}$ is from the partition:

$$\begin{bmatrix} \underline{H}_1 & \mathbf{h} \\ \mathbf{h}^T & h_0 \end{bmatrix} := QHQ. \quad (21)$$

The following remarks are useful for those who are not familiar with the generalized Jacobian of matrix-valued functions (note: $\Pi_{\mathcal{K}_-^n}(X)$ is matrix-valued) studied in [33, 7].

(R1) If we collect all the linear operations $V : \mathcal{S}^n \mapsto \mathcal{S}^n$ defined in (20) in the set:

$$\widehat{\partial}\Phi(X) := \left\{ V \mid V \text{ satisfies (20) for some } \tilde{V} \in \Pi_{\mathcal{S}_+^{|\beta|+1}}(0) \right\},$$

then we have the following property (see [10, Prop. 2.6.4])

$$\partial\Phi(X)H = \widehat{\partial}\Phi(X)H, \quad \forall H \in \mathcal{S}^n.$$

The subtle difference is that it usually hard to characterize $\partial\Phi(X)$, but its image $\partial\Phi(X)H$ can be characterized through the image $\widehat{\partial}\Phi(X)H$, which is sufficient for our theoretical analysis later on.

(R2) When it comes to compute a particular element V , \mathcal{W}_H in (20) can be significantly simplified. For example, it is known that \tilde{V} can be chosen to be the zero operator [8]. All that is needed to characterize \mathcal{W}_H can be done through the spectral decomposition of JXJ :

$$JXJ = P\text{Diag}(\lambda_1, \dots, \lambda_n)P^T,$$

where $\lambda_1 \geq \dots \geq \lambda_n$ are eigenvalues and $PP^T = I$. Define the matrix $\widehat{\Omega} \in \mathcal{S}^n$ by

$$\widehat{\Omega}_{ij} := \frac{\max\{\lambda_i, 0\} + \max\{\lambda_j, 0\}}{|\lambda_i| + |\lambda_j|}, \quad i, j = 1, \dots, n$$

where $0/0$ is defined to be 1. The corresponding V takes the following form:

$$VH = H - P\left(\widehat{\Omega} \circ (P^T(JHJ)P)\right)P^T.$$

This type of matrices have been previously used in implementing Newton's method in [23, 4] for the nearest correlation matrix problem. We will also use this element in our computation.

Next, we give a sufficient condition for any linear operator in $\partial\Phi(X)$ to be positive definite over a subspace. We continue to use the notation in Prop. 2.2.

PROPOSITION 2.3. *Let $X \in \mathcal{S}^n$ be given and the generalized Jacobian $\partial\Phi(X)$ is characterized in Prop. 2.2. Let \mathcal{H}_0 be a subspace in \mathcal{S}^n . For any $H \in \mathcal{S}^n$, let the matrix QHQ have the partition (21). If the following implication holds for all $H \in \mathcal{H}_0$:*

$$\left. \begin{array}{l} W_\gamma^T \underline{H}_1 = 0 \\ \underline{\mathbf{h}} = 0 \\ \underline{h}_0 = 0 \end{array} \right\} \implies H = 0, \quad (22)$$

then every $V \in \partial\Phi(X)$ is positive definite on \mathcal{H}_0 . That is $\langle H, VH \rangle > 0$ for $0 \neq H \in \mathcal{H}_0$.

Proof. The proof involves some heavy calculations between matrices, e.g., multiplications. We omit some of them in order to save space. However, the omitted parts can be directly verified. Noticing that both Q and \overline{W} are orthogonal matrices, we calculate for any $V \in \partial\Phi(X)$ and $H \in \mathcal{H}_0$,

$$\begin{aligned} \langle H, VH \rangle &= \|\overline{W}^T(QHQ)\overline{W}\|^2 - \langle \overline{W}^T(QHQ)\overline{W}, \mathcal{W}_H \rangle \\ &= 2 \{ \|W_\alpha^T \underline{\mathbf{h}}\|^2 + \|W_\alpha^T \underline{H}_1 W_\gamma\|^2 - \langle W_\alpha^T \underline{H}_1 W_\gamma, \Omega_{\alpha\gamma} \circ (W_\alpha^T \underline{H}_1 W_\gamma) \rangle \} \\ &\quad + 2 \{ \|W_\beta^T \underline{H}_1 W_\gamma\|^2 + \|W_\gamma^T \underline{\mathbf{h}}\|^2 + \|W_\gamma^T \underline{H}_1 W_\gamma\|^2 / 2 \} \\ &\quad + \|G_1\|^2 - \langle G_1, \tilde{V}(G_2) \rangle, \end{aligned}$$

where

$$G_1 := \begin{bmatrix} W_\beta^T \underline{H}_1 W_\beta & W_\beta^T \underline{\mathbf{h}} \\ \underline{\mathbf{h}}^T W_\beta & \underline{h}_0 \end{bmatrix}, \quad G_2 := \begin{bmatrix} W_\beta^T \underline{H}_1 W_\beta & 0 \\ 0 & 0 \end{bmatrix}.$$

It is easy to prove that

$$\|G_1\|(\|G_1\| - \|G_2\|) \geq \|W_\beta^T \underline{\mathbf{h}}\|^2 + \frac{1}{2} \underline{h}_0^2. \quad (23)$$

It follows from [8, Eq (17)] that

$$\langle G_1, \tilde{V}(G_2) \rangle \leq \|G_1\| \|G_2\|. \quad (24)$$

Define $\tau_{\max} := \max_{i \in \alpha, j \in \gamma} \Omega_{ij}$. By (19), $0 < \tau_{\max} < 1$. We continue to simplify $\langle H, VH \rangle$.

$$\begin{aligned} \langle H, VH \rangle &\geq 2 \{ \|W_\alpha^T \underline{\mathbf{h}}\|^2 + \|W_\gamma^T \underline{\mathbf{h}}\|^2 + \|W_\beta^T \underline{H}_1 W_\gamma\|^2 + (1 - \tau_{\max}) \|W_\alpha^T \underline{H}_1 W_\gamma\|^2 \} \\ &\quad + \|W_\gamma^T \underline{H}_1 W_\gamma\|^2 + \|G_1\|^2 - \|G_1\| \|G_2\| \quad (\text{by (24)}) \\ &\geq 2 \left\{ \|W_\alpha^T \underline{\mathbf{h}}\|^2 + \|W_\gamma^T \underline{\mathbf{h}}\|^2 + \frac{1}{2} \|W_\beta^T \underline{\mathbf{h}}\|^2 \right\} + \|W_\gamma^T \underline{H}_1 W_\gamma\|^2 \\ &\quad + 2 \{ (1 - \tau_{\max}) \|W_\alpha^T \underline{H}_1 W_\gamma\|^2 + \|W_\beta^T \underline{H}_1 W_\gamma\|^2 \} + \frac{1}{2} \underline{h}_0^2 \quad (\text{by (23)}) \\ &\geq 0. \end{aligned}$$

Hence, the assumption $\langle H, VH \rangle = 0$ would imply

$$\begin{cases} W_\alpha^T \underline{\mathbf{h}} = 0, & W_\beta^T \underline{\mathbf{h}} = 0, & W_\gamma^T \underline{\mathbf{h}} = 0, & \text{and } \underline{h}_0 = 0 \\ W_\alpha^T \underline{H}_1 W_\gamma = 0, & W_\beta^T \underline{H}_1 W_\gamma = 0, & W_\gamma^T \underline{H}_1 W_\gamma = 0, \end{cases}$$

which is equivalent to

$$W^T \underline{\mathbf{h}} = 0, \quad \underline{h}_0 = 0, \quad \text{and} \quad W^T \underline{H}_1 W_\gamma = 0.$$

Because W in (15) is orthogonal, it follows that

$$\underline{\mathbf{h}} = 0, \quad \underline{h}_0 = 0 \quad \text{and} \quad \underline{H}_1 W_\gamma = 0,$$

which in turn by (22) implies that $H = 0$. This proves that $\langle H, VH \rangle \geq 0$ and it equals 0 if and only if $H = 0$. Hence, V is positive definite over \mathcal{H}_0 . \square

2.3. Tangent cone of \mathcal{K}_-^n . Now suppose X belongs to \mathcal{K}_-^n . Let $\mathcal{T}_{\mathcal{K}_-^n}(X)$ denote the tangent cone of \mathcal{K}_-^n at X and $\text{lin}(\mathcal{T}_{\mathcal{K}_-^n}(X))$ be the largest subspace contained in $\mathcal{T}_{\mathcal{K}_-^n}(X)$. We now give a description of this subspace. It follows from (6) that $-JXJ \in \mathcal{S}_+^n$. Then all the eigenvalues of X_1 in (15) are nonpositive. Suppose $r = \text{rank}(X_1)$. Since the eigenvalues in (15) are arranged in nonincreasing order, we must have

$$\underline{\lambda}_1 = \cdots = \underline{\lambda}_{n-r} = 0, \quad 0 > \underline{\lambda}_{n-1-r} \geq \cdots \geq \underline{\lambda}_{n-1}.$$

Using the fact $\mathcal{K}_-^n = -\mathcal{K}_+^n$ and the formula [21, Eq. (23)] for the tangent cone of \mathcal{K}_+^n , we obtain

$$\mathcal{T}_{\mathcal{K}_-^n}(X) = \left\{ Q \left[W \begin{bmatrix} \Sigma_1 & \Sigma_{12} \\ \Sigma_{12}^T & \Sigma_2 \\ \mathbf{a}^T & a_0 \end{bmatrix} W^T \quad \begin{matrix} \mathbf{a} \\ a_0 \end{matrix} \right] Q : \begin{matrix} \Sigma_1 \in \mathcal{S}_+^{n-1-r}, \Sigma_2 \in \mathcal{S}^r \\ \Sigma_{12} \in \mathbb{R}^{(n-1-r) \times r} \\ \mathbf{a} \in \mathbb{R}^{n-1}, a_0 \in \mathbb{R} \end{matrix} \right\}.$$

Therefore, the largest subspace in $\mathcal{T}_{\mathcal{K}_-^n}(X)$ is given by

$$\text{lin}(\mathcal{T}_{\mathcal{K}_-^n}(X)) = \left\{ Q \left[W \begin{bmatrix} 0 & \Sigma_{12} \\ \Sigma_{12}^T & \Sigma_2 \\ \mathbf{a}^T & a_0 \end{bmatrix} W^T \quad \begin{matrix} \mathbf{a} \\ a_0 \end{matrix} \right] Q : \begin{matrix} \Sigma_2 \in \mathcal{S}^r \\ \Sigma_{12} \in \mathbb{R}^{(n-1-r) \times r} \\ \mathbf{a} \in \mathbb{R}^{n-1}, a_0 \in \mathbb{R} \end{matrix} \right\}. \quad (25)$$

In particular, we have

$$\left\{ Q \left[\begin{matrix} 0_{(n-1) \times (n-1)} & \mathbf{a} \\ \mathbf{a}^T & 0 \end{matrix} \right] Q : \mathbf{a} \in \mathbb{R}^{n-1} \right\} \subset \cap_{X \in \mathcal{K}_-^n} \text{lin}(\mathcal{T}_{\mathcal{K}_-^n}(X)). \quad (26)$$

3. Constraint Nondegeneracy. We continue to prepare one more technical result that will eventually lead to the quadratic convergence of Newton's method to be developed soon. This result is about the constraint nondegeneracy of the constraints in (8). We restate the constraints as follows. Define the linear mapping $\mathcal{A} : \mathcal{S}^n \mapsto \mathbb{R}^{n-1}$ by

$$(\mathcal{A}(Y))_i := Y_{ii} - Y_{nn}, \quad i = 1, \dots, n-1, \quad \forall Y \in \mathcal{S}^n.$$

We let $\mathcal{A}^* : \mathbb{R}^{n-1} \mapsto \mathcal{S}^n$ be the adjoint of \mathcal{A} . Then the constraints in (8) become:

$$Y \in \mathcal{K}_-^n \quad \text{and} \quad \mathcal{A}(Y) = 0. \quad (27)$$

The concept of constraint nondegeneracy, which is generalization of the linear independence constraint qualification, has been extensively used by Bonnans and Shapiro [2] for abstract optimization problems. Its prominent role in the stability analysis

of (nonlinear) semidefinite programming has been revealed by Sun [32, 8]. It often implies the quadratic convergence of a class of Newton's method, see [23, 21, 1]. We define this concept below.

DEFINITION 3.1. *We say that constraint nondegeneracy holds at a feasible point X of (27) if*

$$\mathcal{A}\left(\text{lin}(\mathcal{T}_{\mathcal{K}_-^n}(X))\right) = \mathbb{R}^{n-1}. \quad (28)$$

PROPOSITION 3.2. *Constraint nondegeneracy holds at every feasible point of (27).*

Proof. Let X be a feasible point with the decompositions in (14) and (15). It follows from (26) that

$$Y := Q \begin{bmatrix} 0_{(n-1) \times (n-1)} & \mathbf{a} \\ \mathbf{a}^T & 0 \end{bmatrix} \in \text{lin}(\mathcal{T}_{\mathcal{K}_-^n}(X)) \quad \forall \mathbf{a} \in \mathbb{R}^{n-1}.$$

We calculate the diagonals of Y .

$$\begin{aligned} Y_{ii} &= e_i^T Y Q e_i = \text{Tr} \left(Q e_i (e_i^T Q) \begin{bmatrix} 0_{(n-1) \times (n-1)} & \mathbf{a} \\ \mathbf{a}^T & 0 \end{bmatrix} \right) \\ &= e_n^T Q e_i (e_i^T Q) \begin{bmatrix} 2\mathbf{a} \\ 0 \end{bmatrix} \\ &= -\frac{1}{\sqrt{n}} e^T (e_i e_i^T Q) \begin{bmatrix} 2\mathbf{a} \\ 0 \end{bmatrix} \quad (\text{using } Q e_n = -\frac{1}{\sqrt{n}} e) \\ &= -\frac{1}{\sqrt{n}} e_i^T Q \begin{bmatrix} 2\mathbf{a} \\ 0 \end{bmatrix} = -\frac{2}{\sqrt{n}} e_i^T Q_1 \mathbf{a}, \end{aligned}$$

where Q_1 is the submatrix of Q consisting of the first $(n-1)$ columns and e_i is the i th column of the identity matrix in \mathcal{S}^n . We recall the matrix Q in (9) to get Q_1 , whose special structure leads to (note that Q_1 is $n \times (n-1)$ matrix):

$$e_i^T Q_1 = \begin{cases} \tilde{e}_i^T - \frac{1}{\sqrt{n}} \tilde{e}^T + \frac{1}{\sqrt{n+1}} \tilde{e}, & \text{for } i = 1, \dots, n-1 \\ -\frac{1}{\sqrt{n}} \tilde{e}^T, & \text{for } i = n \end{cases}$$

where (to avoid confusion with the base vectors e_i in \mathbb{R}^n) we use \tilde{e}_i to denote it is the i th basis in \mathbb{R}^{n-1} and \tilde{e} is the vector of all ones in \mathbb{R}^{n-1} . Hence, we have

$$Y_{ii} - Y_{nn} = -\frac{2}{\sqrt{n}} (e_i - e_1)^T Q_1 \mathbf{a} = -\frac{2}{\sqrt{n}} \left(\tilde{e}_i + \frac{1}{\sqrt{n+1}} \tilde{e} \right)^T \mathbf{a}, \quad i = 1, \dots, n-1.$$

Therefore, for any $\mathbf{y} \in \mathbb{R}^{n-1}$, the equation $\mathcal{A}(Y) = \mathbf{y}$ becomes

$$\left(\tilde{e}_i + \tilde{e}/(\sqrt{n+1}), \mathbf{a} \right) = -\frac{\sqrt{n}}{2} y_i, \quad i = 1, 2, \dots, n-1.$$

This is a system of $(n-1) \times (n-1)$ linear equations, whose coefficient matrix is nonsingular (the coefficient vectors are linearly independent). Hence, it has a unique solution $\mathbf{a} \in \mathbb{R}^{n-1}$. Therefore, we must have (28) and constraint nondegeneracy holds at X . \square

We have the following consequence of constraint nondegeneracy.

PROPOSITION 3.3. *Let X be a feasible point of (27) and it has the decompositions in (14) and (15), where the eigenvector matrix W is defined. Let γ be defined in (18). For any $\mathbf{h} \in \mathbb{R}^{n-1}$, denote $H := \mathcal{A}^*(\mathbf{h})$ and let H be decomposed as in (21). Then we must have the following implication:*

$$\left. \begin{array}{l} W_\gamma^T \underline{H}_1 = 0 \\ \underline{\mathbf{h}} = 0 \\ \underline{h}_0 = 0 \end{array} \right\} \implies \mathbf{h} = 0. \quad (29)$$

Proof. Let $\mathbf{h} \in \mathbb{R}^{n-1}$ satisfy the left-hand-side condition of (29). We will prove that such \mathbf{h} must belong to $\left\{ \mathcal{A} \left(\text{lin}(\mathcal{T}_{\mathcal{K}_-^n}(X)) \right) \right\}^\perp$, which is $\{0\}$ because of the constraint nondegeneracy (28) at X . Hence, we obtain $\mathbf{h} = 0$ and we would complete the proof. We now prove it is the case.

It follows from (25) that

$$\left\{ QBQ : B \in \text{lin}(\mathcal{T}_{\mathcal{K}_-^n}(X)) \right\} = \left\{ \left[\begin{array}{cc} W \left[\begin{array}{cc} 0 & \Sigma_{12} \\ \Sigma_{12}^T & \Sigma_2 \end{array} \right] W^T & \mathbf{a} \\ \mathbf{a}^T & a_0 \end{array} \right] : \begin{array}{l} \Sigma_2 \in \mathcal{S}^r \\ \Sigma_{12} \in \mathbb{R}^{(n-1-r) \times r} \\ \mathbf{a} \in \mathbb{R}^{n-1}, a_0 \in \mathbb{R} \end{array} \right\}.$$

We have

$$\begin{aligned} \langle \mathbf{h}, \mathcal{A}(B) \rangle &= \langle \mathcal{A}^*(\mathbf{h}), B \rangle = \langle Q\mathcal{A}^*(\mathbf{h})Q, QBQ \rangle && \text{(because } Q^2 = I) \\ &= 2\langle \underline{\mathbf{h}}, \mathbf{a} \rangle + \underline{h}_0 a_0 + \text{Tr} \left(W^T \underline{H}_1 W \left[\begin{array}{cc} 0 & \Sigma_{12} \\ \Sigma_{12}^T & \Sigma_2 \end{array} \right] \right). \end{aligned}$$

We note that the rank r of X_1 in (15) equals $|\gamma|$, where γ is the index set in (18) that contains all negative eigenvalues. The condition $W_\gamma^T \underline{H}_1 = 0$ implies that the last r rows of $W^T \underline{H}_1$ are zeros. This further implies that the last r rows of $(W^T \underline{H}_1 W)$ are zeros, so are the last r columns because of the symmetry of the matrix. Therefore,

$$\text{Tr} \left(W^T \underline{H}_1 W \left[\begin{array}{cc} 0 & \Sigma_{12} \\ \Sigma_{12}^T & \Sigma_2 \end{array} \right] \right) = 0.$$

The other conditions in (29) $\underline{\mathbf{h}} = 0$ and $h_0 = 0$ further imply that $\langle \mathbf{h}, \mathcal{A}(B) \rangle = 0$. This is true for all $B \in \text{lin}(\mathcal{T}_{\mathcal{K}_-^n}(X))$. Hence, we must have $\mathbf{h} \in \left\{ \mathcal{A} \left(\text{lin}(\mathcal{T}_{\mathcal{K}_-^n}(X)) \right) \right\}^\perp$. We established the case. \square

4. Newton-CG Method. In this section, we develop the Newton-CG method for Problem (8), which is equivalent to

$$\min_Y \frac{1}{2} \|Y - \widehat{D}\|^2 \quad \text{s.t.} \quad Y \in \mathcal{K}_-^n \quad \text{and} \quad \mathcal{A}(Y) = 0. \quad (30)$$

Define the Lagrangian function for the problem (30):

$$L(Y; y) := -\frac{1}{2} \|Y - \widehat{D}\|^2 - \langle \mathcal{A}(Y), \mathbf{y} \rangle + \delta_{\mathcal{K}_-^n}(Y),$$

where $\delta_{\mathcal{K}_-^n}(\cdot)$ is the indicator function of \mathcal{K}_-^n . The Lagrangian dual function is

$$\begin{aligned}\theta(\mathbf{y}) &:= \min_{Y \in \mathcal{S}^n} L(Y; \mathbf{y}) \\ &= \min_{Y \in \mathcal{S}^n} \frac{1}{2} \|Y - (\widehat{D} + \mathcal{A}^* \mathbf{y})\|^2 + \delta_{\mathcal{K}_-^n}(Y) - \frac{1}{2} \|\widehat{D} + \mathcal{A}^* \mathbf{y}\|^2 + \frac{1}{2} \|\widehat{D}\|^2 \\ &= \frac{1}{2} \|\Pi_{\mathcal{K}_-^n}(\widehat{D} + \mathcal{A}^* \mathbf{y}) - (\widehat{D} + \mathcal{A}^* \mathbf{y})\|^2 - \frac{1}{2} \|\widehat{D} + \mathcal{A}^* \mathbf{y}\|^2 + \frac{1}{2} \|\widehat{D}\|^2 \\ &= -\frac{1}{2} \|\Pi_{\mathcal{K}_-^n}(\widehat{D} + \mathcal{A}^* \mathbf{y})\|^2 + \frac{1}{2} \|\widehat{D}\|^2.\end{aligned}$$

The Lagrangian dual problem in the form minimization is defined by

$$\min_{\mathbf{y} \in \mathbb{R}^{n-1}} \Theta(\mathbf{y}) := -\theta(\mathbf{y}) = \frac{1}{2} \|\Pi_{\mathcal{K}_-^n}(\widehat{D} + \mathcal{A}^* \mathbf{y})\|^2 - \frac{1}{2} \|\widehat{D}\|^2. \quad (31)$$

Since \mathcal{K}_-^n is a closed and convex cone, $\Theta(\cdot)$ is convex and continuously differentiable (see [16, Chapter IV, Example 2.1.4]). We also note that $\mathcal{S}_+^n \subseteq \mathcal{K}_-^n$ and the negative identity matrix satisfies the constraints in (30). This implies that the generalized Salter condition¹ holds for the constraints in (30). It follows from the general results [15, Prop. 2.20, Prop. 4.11] that the dual function $\Theta(\cdot)$ is coercive (i.e., $\Theta(\mathbf{y}) \rightarrow \infty$ as $\|\mathbf{y}\| \rightarrow \infty$). Therefore the dual problem (31) must admit an optimal solution and the first-order optimality condition is

$$F(\mathbf{y}) := \nabla \Theta(\mathbf{y}) = \mathcal{A} \left(\Pi_{\mathcal{K}_-^n}(\widehat{D} + \mathcal{A}^*(\mathbf{y})) \right) = 0. \quad (32)$$

It follows from the projection formula (11) of Gaffke and Mathar that $F(\mathbf{y})$ is *strongly semismooth*² because it is a composition of linear mappings and $\Pi_{\mathcal{S}_+^n}(\cdot)$, which is known to be strongly semismooth [33, 7]. Now it becomes natural to develop the semismooth Newton method for the semismooth equation (32): Given $\mathbf{y}^0 \in \mathbb{R}^{n-1}$ and letting $\ell := 0$, compute $M_\ell \in \partial F(\mathbf{y}^\ell)$ and

$$\mathbf{y}^{\ell+1} := \mathbf{y}^\ell - M_\ell^{-1} F(\mathbf{y}^\ell), \quad \ell = 0, 1, 2, \dots \quad (33)$$

Since F is the gradient of Θ , ∂F is often called the generalized Hessian of Θ .

Let \mathbf{y}^* be the optimal solution of the dual problem (31), then the optimal solution Y^* of the original problem (30) is given by

$$Y^* = \Pi_{\mathcal{K}_-^n}(\widehat{D} + \mathcal{A}^*(\mathbf{y}^*)). \quad (34)$$

We now study the structure of $\partial F(\mathbf{y})$ and its nonsingularity at \mathbf{y} . Following the chain rule of the generalized Jacobian, we have

$$\partial F(\mathbf{y}) \mathbf{h} = \mathcal{A} \left(\partial \Pi_{\mathcal{K}_-^n}(X)(\mathcal{A}^* \mathbf{h}) \right), \quad \forall \mathbf{h} \in \mathbb{R}^{n-1}, \quad (35)$$

¹This is because \mathcal{A} is linear independent, the negative identity matrix is in the interior of \mathcal{K}_-^n and $\mathcal{A}(I) = 0$.

²A (locally) Lipschitz function $\Phi : \mathbb{R}^m \mapsto \mathbb{R}^\ell$ is said to be strongly semismooth at $x \in \mathbb{R}^m$ if (i) Φ is directionally differentiable at x , and (ii) for any $V \in \partial \Phi(x+h)$,

$$\Phi(x+h) - \Phi(x) - Vh = o(\|h\|^2), \quad h \in \mathbb{R}^m,$$

where $\partial \Phi(x)$ denotes the generalized Jacobian of Φ at x in the sense of Clarke [10, Sect. 2.6].

where $X := \widehat{D} + \mathcal{A}^* \mathbf{y}$ and $\partial \Pi_{\mathcal{K}_-^n}(X)$ is characterized in Prop. 2.2. We have the following result.

PROPOSITION 4.1. *Every matrix element in $\partial F(\mathbf{y})$ is positive definite for any $\mathbf{y} \in \mathbb{R}^{n-1}$. Consequently, the Newton equation in (33) is well defined.*

Proof. Let M be an arbitrary matrix in $\partial F(\mathbf{y})$. It follows from the chain rule in (35) that there exists $V \in \partial \Pi_{\mathcal{K}_-^n}(X)$ such that

$$M\mathbf{h} = \mathcal{A} \left(V(\mathcal{A}^* \mathbf{h}) \right), \quad \forall \mathbf{h} \in \mathbb{R}^{n-1}.$$

This implies

$$\langle \mathbf{h}, M\mathbf{h} \rangle = \langle \mathcal{A}^* \mathbf{h}, V(\mathcal{A}^* \mathbf{h}) \rangle.$$

Because \mathcal{A} is linearly independent, the positive definiteness of M is implied by that of V restricted to the subspace $\mathcal{H}_0 := \{H = \mathcal{A}^* \mathbf{h} : \mathbf{h} \in \mathbb{R}^{n-1}\}$. We have proved in Prop. 2.3 that a sufficient condition for V being positive definite on \mathcal{H}_0 is that Condition (22) holds for all $H \in \mathcal{H}_0$. We now prove it is the case.

We now note that the matrix $X = \widehat{D} + \mathcal{A}^* \mathbf{y}$ and $Y := \Pi_{\mathcal{K}_-^n}(X)$ have the following relationship from (12):

$$-JYJ = \Pi_{\mathcal{S}_+^n}(-JXJ) = Q \begin{bmatrix} \Pi_{\mathcal{S}_+^{n-1}}(-\underline{X}_1) & 0 \\ 0 & 0 \end{bmatrix} Q, \quad (36)$$

where \underline{X}_1 is from the decomposition in (13). Following a similar calculation as in (13), we decompose JYJ by

$$JYJ = Q \begin{bmatrix} \underline{Y}_1 & 0 \\ 0 & 0 \end{bmatrix} Q.$$

The relation in (36) implies that

$$-\underline{Y}_1 = \Pi_{\mathcal{S}_+^{n-1}}(-\underline{X}_1),$$

Equivalently, we have $\underline{Y}_1 = \Pi_{\mathcal{S}_-^{n-1}}(\underline{X}_1)$. That is to say that the negative eigenvalues of \underline{Y}_1 are the negative eigenvalues of \underline{X}_1 . Furthermore, they share common eigenvectors for those negative eigenvalues. Those eigenvectors are contained in W_γ with γ being defined in (18). Therefore, Condition (22) is just Condition (29) with $H = \mathcal{A}^* \mathbf{h}$. We already proved that Condition (29) is automatically satisfied because constraint nondegeneracy holds at Y by Prop. 3.2 and Prop. 3.3. Therefore, V is positive definite on \mathcal{H}_0 and consequently M is positive definite. \square

A direct consequence of Prop. 4.1 is that the Newton method in (33) is quadratically convergent. We state this result below.

PROPOSITION 4.2. *Let \mathbf{y}^* be the optimal solution of the dual problem. Then the generalized Jacobian of $\partial F(\mathbf{y}^*)$ is positive definite. Consequently, \mathbf{y}^* is the unique solution of the dual problem (31) and the semismooth Newton method (33) is quadratically convergent provided that the initial point \mathbf{y}^0 is sufficiently close to \mathbf{y}^* .*

Proof. The positive definiteness of $\partial F(\mathbf{y}^*)$ is just the direct consequence of Prop. 4.1 at \mathbf{y}^* . The generalized Jacobian $\partial F(\mathbf{y}^*)$ has the maximal rank because every element in it is nonsingular (i.e., positive definite). The inverse function theorem of Clarke [10, Thm. 7.1.1] and the convexity of the dual problem (31) ensure that \mathbf{y}^*

is the unique solution. The quadratic convergence follows the convergence result for semismooth Newton method of Qi and Sun [25, Thm. 3.2] provided that the initial point \mathbf{y}^0 is close enough to \mathbf{y}^* . \square

We finish this subsection by an important remark. Estimating the Newton matrix M_ℓ in (33) has a computational complexity $O(n^4)$, which is prohibitive when n is large. However, estimating the matrix and vector product $M_\ell \mathbf{h}$ is much cheaper and the sparse structure in V_ℓ can be exploited. Moreover, the matrix M_ℓ is always positive definite (Prop. 4.1). Those are the key reasons why the conjugate gradient method was originally suggested by Qi and Sun [23] to solve such (positive definite) Newton equations. Therefore, the method is termed as Newton-CG method, which has found applications in other situations [38, 21]. The Newton method in (33) is just a local version. A globalization strategy needs to be used in order to ensure its global convergence. In our numerical implementation, we used the one via Armijo line search proposed in [23, Alg. 5.1], which is adapted to our case below.

ALGORITHM 4.3. Semismooth Newton-CG Method: $NCG(\widehat{D})$.

Input: an $n \times n$ distance matrix \widehat{D} .

Output: an $n \times n$ Euclidean distance matrix D^* .

(S.0) Given $\mathbf{y}^0 \in \mathbb{R}^{n-1}$, $\eta \in (0, 1)$, $\sigma \in (0, 1)$, $\kappa_1 \in (0, 1)$, $\kappa_2 \in (1, \infty)$, $\kappa_3 \in (1, \infty)$, and $\delta \in (0, 1)$. Let $\epsilon > 0$ be the given termination tolerance. Let $\ell := 0$.

(S.1) Select an element $M_\ell \in \partial \mathcal{F}(\mathbf{y}^\ell)$, compute $t_\ell := \min\{\kappa_1, \kappa_2 \|\nabla \theta(\mathbf{y}^\ell)\|\}$, and apply the CG method [17] starting with the zero vector as the initial search direction to

$$(M_\ell + t_\ell I) \Delta \mathbf{y} = -\nabla \theta(\mathbf{y}^\ell) \quad (37)$$

to find a search direction $\Delta \mathbf{y}^\ell$ such that

$$\|\nabla \theta(\mathbf{y}^\ell) + (M_\ell + t_\ell I) \Delta \mathbf{y}^\ell\| \leq \eta_j \|\nabla \theta(\mathbf{y}^\ell)\|, \quad (38)$$

where $\eta_\ell := \min\{\eta, \kappa_3 \|\nabla \theta(\mathbf{x}^\ell)\|\}$.

(S.2) Let k_ℓ be the smallest nonnegative integer k such that

$$\theta(\mathbf{y}^k + \delta^k \Delta \mathbf{y}^\ell) - \theta(\mathbf{y}^\ell) \leq \sigma \delta^k \langle \nabla \theta(\mathbf{y}^\ell), \Delta \mathbf{y}^\ell \rangle.$$

Set $\tau_\ell := \delta^{k_\ell}$ and $\mathbf{y}^{\ell+1} := \mathbf{y}^\ell + \tau_\ell \Delta \mathbf{y}^\ell$.

(S.3) If $\|\nabla \theta(\mathbf{y}^{\ell+1})\| \leq \epsilon$. Stop. Otherwise, replace ℓ by $\ell + 1$ and go to (S.1).

(S.4) Upon termination, output

$$Y^{\ell+1} := \Pi_{\mathcal{K}^n}(\widehat{D} + \mathcal{A}^* \mathbf{y}^{\ell+1}) \quad \text{and} \quad \mathbf{a} := \text{diag}(Y^{\ell+1}).$$

The final Euclidean distance matrix D^* is given by

$$D^* := Y^{\ell+1} - \frac{1}{2}(\mathbf{a} \mathbf{e}^T + \mathbf{e} \mathbf{a}^T).$$

We end this section by explaining why the output D^* must be a true Euclidean distance matrix. After the termination test, $Y^{\ell+1}$ is almost negative semidefinite. It follows from the characterization (5) that the matrix D^* thus obtained in Step 4 must be Euclidean. If the diagonals of $Y^{\ell+1}$ are all equal (note: due to practical computational errors, $\mathcal{A}(Y^{\ell+1}) = 0$ may not be exactly satisfied), D^* is Euclidean also follows from (7).

5. Application to LLE. Alg. 4.3 has a number of direct applications. For example, it can be directly used to address the indefiniteness in ISOMAP. However, in this section we include a novel application to LLE [26], a very-known method in nonlinear dimensionality reduction. The main purpose of LLE is to reveal the hidden low-dimensional structures among the given high-dimensional data $X := [\mathbf{x}_1, \dots, \mathbf{x}_n]$ with \mathbf{x}_i being in \mathbb{R}^N . Our main interest here is when X has missing values, to which LLE has no obvious extensions. This section has three parts. Firstly, we briefly review $\text{LLE}(X)$ when X is completely known. We then describe $\text{LLE}(D)$ when the Euclidean distances among X are known. Both $\text{LLE}(X)$ and $\text{LLE}(D)$ have been documented in [27]. Our key result Prop. 5.1 shows that the local Gram matrix in LLE can be obtained through a local Euclidean distance matrix. This result motivated us to use the developed Newton-CG method to make the local distance matrix, which is not Euclidean due to the missing values, to be Euclidean. In this way, we can handle the missing value case. Finally, we numerically demonstrate the efficiency of our method, denoted as LLEMDS.

5.1. LLE(X): LLE from coordinates. Suppose we are given a set of the coordinates of n points $X := [\mathbf{x}_1, \dots, \mathbf{x}_n]$ in the high dimensional space \mathbb{R}^N . The purpose is to embed the n points in a low dimensional space \mathbb{R}^r (e.g., $r = 2, 3$ for visualization) with the local structure of the original points being preserved. LLE has three steps to accomplish this task. The **first step** (Neighborhood search) is to construct the neighborhood index set for each point \mathbf{x}_i . Let \mathcal{N}_i denote the index set of the neighboring points of \mathbf{x}_i (say through the K -nearest neighbors). The **second step** (Weight matrix construction) is to construct a weight vector $\mathbf{w}_i \in \mathbb{R}^{|\mathcal{N}_i|}$ for each point \mathbf{x}_i to be the solution of the following linear equations:

$$G_i \mathbf{w}_i = e \quad \text{and} \quad \sum_{j \in \mathcal{N}_i} \mathbf{w}_i(j) = 1, \quad (39)$$

where $G_i \in \mathcal{S}^{|\mathcal{N}_i|}$ is the inner-product matrix (Local Gram matrix indexed by \mathcal{N}_i):

$$G_i(j, \ell) = \langle \mathbf{x}_j - \mathbf{x}_i, \mathbf{x}_\ell - \mathbf{x}_i \rangle, \quad j, \ell \in \mathcal{N}_i. \quad (40)$$

The geometric meaning of (39) is that the point \mathbf{x}_i is best linearly approximated through its neighboring points in \mathcal{N}_i .

The **third step** (Computing embedding points) is to find the bottom $(r + 1)$ eigenvectors $\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_{r+1}$ of the sparse and symmetric matrix $M \in \mathcal{S}^n$ defined by

$$M := (I - W)^T (I - W), \quad (41)$$

where the weigh matrix $W \in \mathbb{R}^{n \times n}$ is defined as follows. The i th row of W , denoted as $W_{i:}$ is given by

$$W_{i:}(j) := \begin{cases} \mathbf{w}_i(j), & \text{for } j \in \mathcal{N}_i \\ 0, & \text{otherwise.} \end{cases}$$

It is noted that the bottom eigenvector \mathbf{v}_1 is constant $\mathbf{v}_1 = \frac{1}{\sqrt{n}}e$. Hence, the embedding points in \mathbb{R}^r are given by the remaining r eigenvectors $\mathbf{v}_2, \dots, \mathbf{v}_{r+1}$. The detailed description including regularization of LLE can be found in [27, Sect. 4 (Implementation)]. Its Matlab code is freely available online. Our focus is on the input X having some values missing.

5.2. LLE(D): LLE from distances. Suppose now that the only information available is a true Euclidean distance matrix D . Step 1 and Step 3 of LLE will go through without any difficulties as they do not involve any coordinates of existing points. But we need to construct the local Gram matrix G_i for each point \mathbf{x}_i . It can be constructed from D as follows.

Let $D_i \in \mathcal{S}^{1+|\mathcal{N}_i|}$ be the principal matrix of D indexed by $\{i\} \cup \mathcal{N}_i$, with the first row (and column) of D_i consisting of distances from point \mathbf{x}_i to other points indexed by \mathcal{N}_i . Since any principal matrix of an Euclidean matrix must also be Euclidean, D_i is Euclidean. cMDS in (1) is able to generate $(1 + |\mathcal{N}_i|)$ points labelled as $\{\mathbf{z}_0, \mathbf{z}_1, \dots, \mathbf{z}_{|\mathcal{N}_i|}\}$:

$$B_i := -\frac{1}{2}JD_iJ = Z^T Z \quad \text{with } Z := [\mathbf{z}_0, \mathbf{z}_1, \dots, \mathbf{z}_{|\mathcal{N}_i|}] \text{ and } J := I - \frac{1}{1 + |\mathcal{N}_i|}\tilde{e}\tilde{e}^T, \quad (42)$$

where I is the identity matrix in $\mathcal{S}^{1+|\mathcal{N}_i|}$ and \tilde{e} is the column vector of all ones in $\mathbb{R}^{1+|\mathcal{N}_i|}$. Here, \mathbf{z}_0 plays the same role as \mathbf{x}_i in Step 2 of LLE and the rows (and columns) of B_i are indexed from 0 to $|\mathcal{N}_i|$. The local Gram matrix $G_i \in \mathcal{S}^{|\mathcal{N}_i|}$ is then given by

$$G_i(j, \ell) = \langle \mathbf{z}_j - \mathbf{z}_0, \mathbf{z}_\ell - \mathbf{z}_0 \rangle = B_i(0, 0) - B_i(j, 0) - B_i(\ell, 0) + B_i(j, \ell). \quad (43)$$

Once G_i is obtained, LLE can be implemented as before and is described in [27, Sect. 5.1]. We now prove that there is no need to form the matrix B_i in order to calculate the local Gram matrix G_i in (43).

PROPOSITION 5.1. *Let D_i be partitioned as follows:*

$$D_i = \begin{bmatrix} 0 & \mathbf{s}_i^T \\ \mathbf{s}_i & S_i \end{bmatrix} \quad \text{with } S_i \in \mathcal{S}^{|\mathcal{N}_i|}, \mathbf{s}_i \in \mathbb{R}^{|\mathcal{N}_i|}.$$

Then we have

$$G_i = -\frac{1}{2}(S_i - (\mathbf{s}_i e^T + e \mathbf{s}_i^T)), \quad (44)$$

where e is the column vector of all ones in $\mathbb{R}^{|\mathcal{N}_i|}$.

Proof. We let $\mathbf{s} := (1, 0, \dots, 0)^T \in \mathbb{R}^{1+|\mathcal{N}_i|}$ be the first coordinate in $\mathbb{R}^{1+|\mathcal{N}_i|}$ and define the matrix

$$J_s := I - \tilde{e}\mathbf{s}^T,$$

where I is the identity matrix in $\mathcal{S}^{1+|\mathcal{N}_i|}$. On the one hand, following (43) we have

$$\begin{aligned} J_s B_i J_s^T &= J_s Z^T Z J_s^T = (Z - \mathbf{z}_0 \tilde{e}^T)^T (Z - \mathbf{z}_0 \tilde{e}^T) \\ &= \begin{bmatrix} 0 \\ (\mathbf{z}_1 - \mathbf{z}_0)^T \\ \vdots \\ (\mathbf{z}_{|\mathcal{N}_i|} - \mathbf{z}_0)^T \end{bmatrix} [0, \mathbf{z}_1 - \mathbf{z}_0, \dots, \mathbf{z}_{|\mathcal{N}_i|} - \mathbf{z}_0] = \begin{bmatrix} 0 & 0 \\ 0 & G_i \end{bmatrix}. \end{aligned} \quad (45)$$

On the other hand, we have

$$J_s B_i J_s^T = J_s J \left(-\frac{1}{2} D_i \right) J J_s^T = -\frac{1}{2} J_s D_i J_s^T, \quad (46)$$

where we used the fact that $\tilde{e}^T \mathbf{s} = 1$ and

$$J_s J = (I - e\mathbf{s}^T) \left(I - \frac{1}{1 + |\mathcal{N}_i|} \tilde{e}\tilde{e}^T \right) = I - \tilde{e}\mathbf{s}^T = J_s.$$

Taking into consideration of the partition in D_i , we have $\mathbf{s}^T D_i \mathbf{s} = 0$, which leads to

$$\begin{aligned} J_s D_i J_s^T &= (I - \tilde{e}\mathbf{s}^T) \begin{bmatrix} 0 & \mathbf{s}_i^T \\ \mathbf{s}_i & S_i \end{bmatrix} (I - \mathbf{s}\tilde{e}^T) \\ &= \begin{bmatrix} 0 & \mathbf{s}_i^T \\ \mathbf{s}_i & S_i \end{bmatrix} - \left(\begin{bmatrix} 0 \\ \mathbf{s}_i \end{bmatrix} \tilde{e}^T + \tilde{e}[0, \mathbf{s}_i^T] \right). \end{aligned} \quad (47)$$

The claimed result follows from (45), (46) and (47). \square

It follows from (44) that the local Gram matrix G_i defined in (43) is independent of the choice of the embedding points \mathbf{z}_i . If the Euclidean distance matrix D is from the set of points in X in Subject. 5.1, then the local Gram matrix G_i in (40) and G_i in (43) are same because $\{\mathbf{x}_i\}$ are just one set of embedding points of D . Therefore, we have the following corollary.

COROLLARY 5.2. *If D is given by the set of points $X = [\mathbf{x}_1, \dots, \mathbf{x}_n]$ with $\mathbf{x}_i \in \mathbb{R}^N$, then $\text{LLE}(D)$ and $\text{LLE}(X)$ will generate a same set of embedding points.*

The difficulty arises when the local distance matrix D_i is not Euclidean due to the missing values in X . For this case, the geometric property of the linear equations (39) in computing the weight vectors do not hold any more, hence breaking the fundamental principal of LLE that is to construct the best linear approximation to \mathbf{x}_i through its neighbors. This non-Euclidean issue has not been well addressed in literature. To indicate that D might not be Euclidean, we use \hat{D} to differentiate it from D being Euclidean. Fortunately, Cor. 5.2 immediately suggests to make each local distance matrix \hat{D}_i Euclidean. Since the size of the resulting matrix optimization problem will be small, equal to the neighborhood size $|\mathcal{N}_i|$, we will end up with a very fast algorithm, which is denoted as LLEMDS.

ALGORITHM 5.3. LLEMDS (\hat{D}).

Input: $n \times n$ distance matrix \hat{D} (pairwise distances from the available data in X).

Output: n embedding points in \mathbb{R}^r : $\mathbf{y}_i \in \mathbb{R}^r$, $i = 1, \dots, n$.

Step 1. For each $i = 1, \dots, n$, find neighborhood index set \mathcal{N}_i through K -nearest neighbors.

Step 2. For each i , draw the neighboring matrix $\hat{D}_i \in \mathcal{S}^{1+|\mathcal{N}_i|}$ from \hat{D} . Apply NCG(\hat{D}_i) in Alg. 4.3 to output a true Euclidean distance matrix D_i . Compute the local Gram matrix G_i from D_i by (44).

Step 3. Get the weight matrix W by solving the equations (39) for $i = 1, \dots, n$.

Step 4. Compute the bottom $(r + 1)$ eigenvector of the matrix M in (41). The embedding points $\mathbf{y}_i \in \mathbb{R}^r$, $i = 1, \dots, n$ are formed by the $(r + 1)$ eigenvectors except the first one.

Comparing to the original LLE, the extra computation in LLEMDS is in Step 2, which is to compute a true Euclidean distance matrix D_i . When the neighborhood size is small, the Newton-CG method developed in this paper allows us fast computing of D_i . For example, when the K -nearest neighbors is used for \mathcal{N}_i , the complexity of NCG is about $O(K^3)$, which is repeated n times. This results in an overall complexity of $O(nK^3)$ in Step 2, comparable to the original LLE. We will demonstrate its numerical efficiency in the next subsection.

5.3. Numerical examples. In this part, we report the numerical experiments of Alg. 5.3 on several popular image data sets. We conduct two groups of experiments. One is for problems that have deterministic missing-values. The purpose of this test is, on the one hand, to ensure that the reported results are reproducible, and on other hand, to deliberately fail the imputation method in such a way that if it is used to remove the dimensions that have missing values, then there would be no data left for use. The second group consist of problems with random missing-values. Below we detail the two experiments, which were conducted in the `Matlab` R2014a on a Windows 7 desktop with a 64-bit operating system having Intel Core 2 Duo CPU of 3.16 GHz and 4.0 GB of RAM.

(a) Experiments with deterministic missing-values. Recall the data size of X : $[N, n] = \text{size}(X)$ (we use Matlab notation for simplicity), with N being the dimension of each data points \mathbf{x}_i in X , $i = 1, \dots, n$. Suppose that the first m points in X have missing values and the remaining points do not. We further assume that each point has L missing values. More precisely, we set $m = \text{floor}(N/L) - 1$. For \mathbf{x}_i , $i = 1, \dots, m-1$, its L components from the position $L(i-1)+1$ are removed. For \mathbf{x}_m , its components from $L(m-1)+1$ are removed. Therefore, in total, we have removed N values from X and the missing values occur in every dimension. Consequently, if the imputation method is to be used, all dimensions would be removed and there would be no data left for use.

The first data tested is `Teapots`³ data, which has $n = 400$ points, each represents a color picture ($N = 76 \times 101 \times 3 = 23028$) of a teapot. The pictures were taken by a camera circulating around the teapot. Despite the data being in a very high dimensional space, they can be embedded ideally along a circle in 2 dimensions. Fig. 5.1a is embedded by LLE on the complete data. Now we set $m = 93$ (meaning more than 23% of the data points having missing values), $L = 250$ (meaning more than 1% of the values in each point are missing), and $K = 4$ (the neighborhood size used in LLE). The original LLE would not be able to embed those data with missing values. Fig. 5.1b is the embedding generated by our method LLEMDS. It can be seen that it has a similar structure as in Fig. 5.1a, but less smooth on the circle due to the lack of information of those missing values.

The second example is the digit 1 data set from MNIST database [18]. It has $n = 1135$ 8-bit grayscale images and each image has a resolution of 28×28 pixels (resulting in $N = 784$). Fig. 5.1c is the LLE embedding for the complete data set, whereas Fig. 5.1d is the embedding of LLEMDS with $m = 14$ (about 1% of the data points have missing values), $L = 80$ (more than 10% of the values of each point are missing) and $K = 6$. It can be seen that Fig. 5.1d has sharp edges compared to its LLE counterpart. This effect is due to the missing values. But the key features of slantness and the thickness of the digit have been captured by LLEMDS despite the missing values. The third example is the face image data `Face698` [34], which has $n = 698$ images of faces with different poses (up-down, left-right) and each image has 64×64 pixels ($N = 4096$). Fig. 5.1e is the embedding by LLE on the complete data and Fig 5.1f is the embedding by LLEMDS for the incomplete data where $m = 82$ (more than 11% of the data points have missing values), $L = 50$ (more than 1% values of each point are missing) and $K = 12$. It can be seen that LLEMDS is still capable of capturing the key features among the images without having to remove any points that have missing values.

(b) Experiments with random missing-values. We note that there are two

³available from <http://www.cc.gatech.edu/~lsong/code.html>.

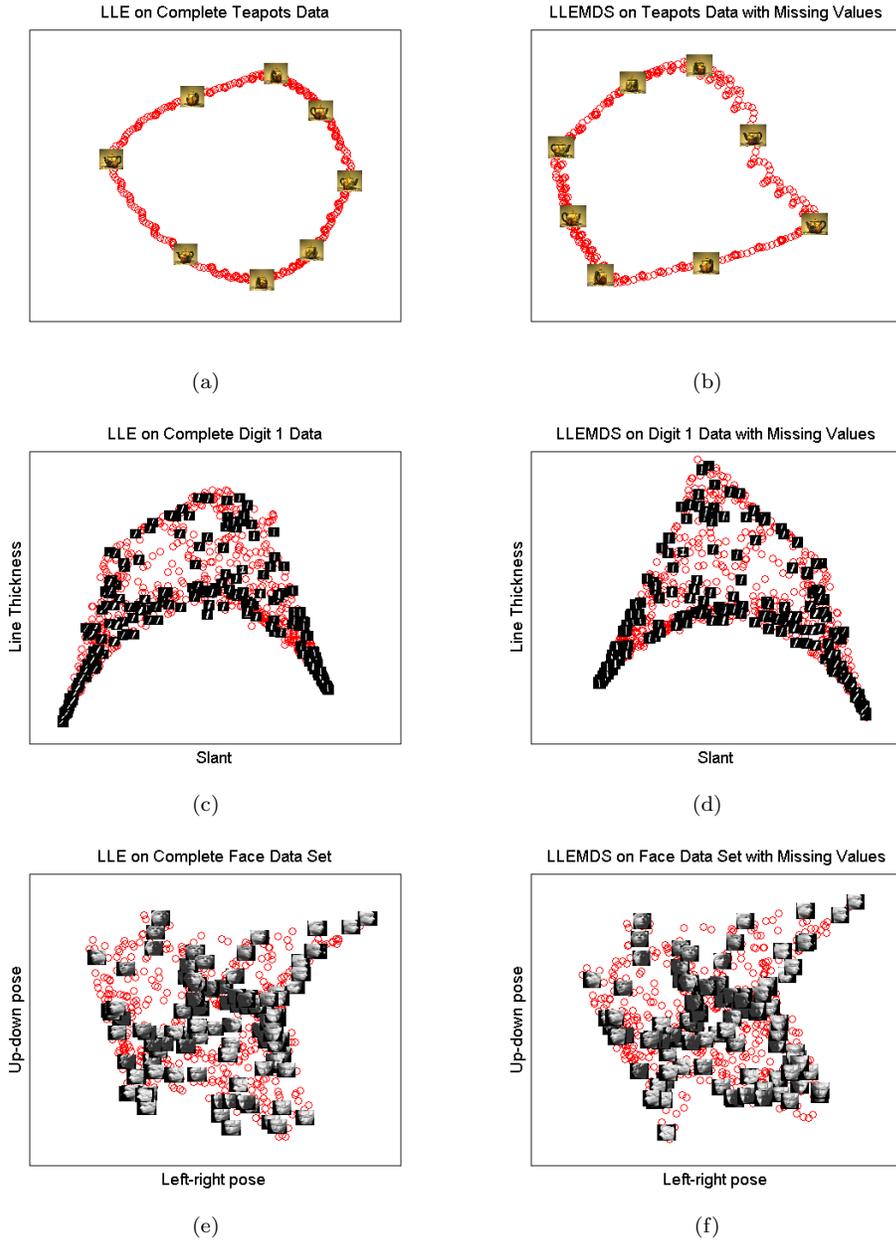


FIG. 5.1. Various data sets embedded by LLE for complete data and by LLEMDS with missing values.

indices (both in percentages) that decide the amount of data missing in X . One is the percentage of the data points that have missing values. We denote it by p_D (i.e., $p_D = m/n$). The other is the percentage of the number of the missing values in each point. We denote it by p_L (i.e., $p_L = L/N$). The tests in part (a) have chosen the missing values deterministically and kept one of the indices at about 1% level, which is enough to fail the imputation method. One of the purposes in this part is

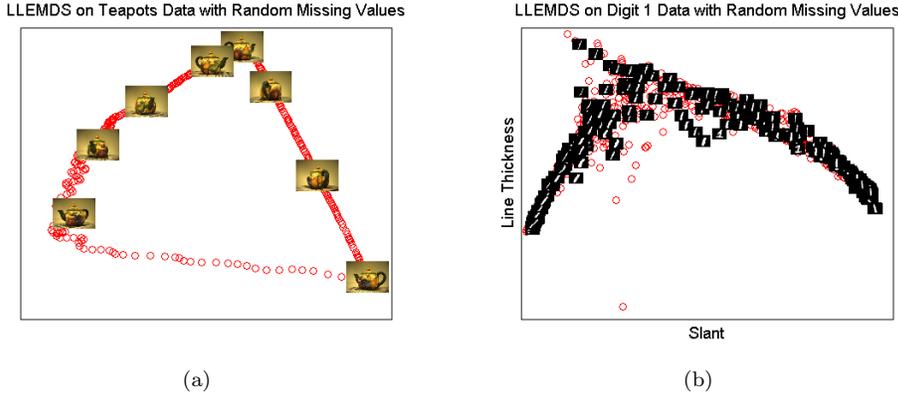


FIG. 5.2. (a) Teapots data embedded by LLEMDS with missing values $p_D = 10\%$, $p_L = 10\%$ and $K = 4$. (b) Digit 1 data embedded by LLEMDS with missing values $p_D = 20\%$, $p_L = 10\%$ and $K = 6$.

to demonstrate that LLEMDS still returns meaningful embeddings when both indices have higher values (e.g., at 10% level).

We use the Matlab built-in function `datasample.m` to randomly choose the missing values in the data matrix X , with the `Replace` option being `false`. For example, for the `Teapots` data, we set $p_D = 0.1$, $p_L = 0.1$, and $K = 4$. This results in about 40 points (i.e., $40 = p_D \times 400$ and randomly chosen) having missing values and each point has about $L = 2300$ (i.e., $L = p_L \times N$ and randomly chosen) missing values. A typical embedding is shown in Fig. 5.2a, which still yielded the correct order (the key feature) of the 400 pictures of the data. But the shape is more a triangle rather than a circle. This is also because of the large amount of data missing. For the `Digit 1` data, we set $p_D = 0.2$, $p_L = 0.1$, and $K = 6$. A typical embedding is shown in Fig. 5.2b, which also revealed the key features (line thickness and slantness) of 1135 images of Digit 1 despite the fact that 20% of the data points each has 10% missing values.

The second purpose of this part of the experiments is to show that LLEMDS is still fast enough when compared to the original LLE. In Table 1, we report our experiments on 8 data sets: `Teapots`, `Face698`, `Face1965` [26], and the MNIST data sets. We combined a few of the digits data sets in MNIST to create new and larger test data. For example, `Digit1to5` is the set including test data of digits from 1 to 5, totally $n = 3969$ data points. For each data set, we tested three scenarios: $p_D = 5\%$, 10% , and 20% with $K = 8$ or $K = 16$ and $p_L = 10\%$. In Table 1, we report the `cpu` times used by LLE when X has no missing values, by LLEMDS when X has random missing values governed by p_D and p_L ; and by `cpu(D)`, which is the time used for random sampling and computing the pairwise Euclidean distances from X when all the missing values are discarded. As seen from Table 1, LLEMDS is, on average, only 2 or 3 times slower than the original LLE. In particular, when the number of data points gets bigger, the time difference used by the two methods gets smaller. For example, for the data set `Digit6to9` ($n = 5051$, $p_D = 0.2$, $K = 8$), `LLE(X)` used about 5 seconds (i.e., 5.33s) and `LLEMDS(D)` used less than 8 seconds (i.e., 7.73s). Given that the original LLE is very fast, our method is still very competitive and is able to deal with the case of missing values. This shows that our Newton-CG method is efficient enough not to drastically slow down the original LLE.

TABLE 1

Comparison on *cpu* (in seconds) between LLE and LLEMDS on 8 data sets. LLE(X): *cpu* time by LLE when X has no missing values; LLEMDS(\hat{D}): *cpu* time by LLEMDS when X has no missing values; *cpu*(\hat{D}): *cpu* time for random sampling and computing the pairwise Euclidean distances from X when the missing values are discarded. Time reported is the average over 10 runs.

Data	p_D	$K = 8, p_L = 0.1$			$K = 16, p_L = 0.1$		
		LLE(X)	LLEMDS(\hat{D})	<i>cpu</i> (\hat{D})	LLE(X)	LLEMDS(\hat{D})	<i>cpu</i> (\hat{D})
Teapots $N = 23028$ $n = 400$	0.05	1.85E-01	5.04E-01	7.82E+00	1.96E-01	1.08E+00	6.78E+00
	0.1	1.84E-01	5.07E-01	1.17E+01	1.79E-01	1.49E+00	1.17E+01
	0.2	1.74E-01	7.02E-01	2.30E+01	1.83E-01	1.87E+00	2.33E+01
Face698 $N = 4096$ $n = 698$	0.05	1.80E-01	6.01E-01	3.08E+00	2.33E-01	8.56E-01	3.14E+00
	0.1	1.83E-01	6.27E-01	5.19E+00	2.46E-01	1.01E+00	5.33E+00
	0.2	1.71E-01	6.42E-01	9.52E+00	2.32E-01	1.30E+00	9.67E+00
Face1965 $N = 560$ $n = 1965$	0.05	4.89E-01	1.68E+00	3.70E+00	1.27E+00	3.16E+00	3.61E+00
	0.1	4.90E-01	1.69E+00	6.29E+00	1.28E+00	3.55E+00	6.20E+00
	0.2	4.92E-01	1.73E+00	1.29E+01	1.26E+00	4.63E+00	1.25E+01
Digit 1 $N = 784$ $n = 1135$	0.05	2.83E-01	9.74E-01	1.53E+00	5.59E-01	1.41E+00	1.51E+00
	0.1	2.84E-01	9.77E-01	2.61E+00	5.69E-01	1.56E+00	2.58E+00
	0.2	2.97E-01	9.82E-01	4.58E+00	5.76E-01	1.89E+00	4.53E+00
Digit 9 $N = 784$ $n = 1009$	0.05	3.25E-01	9.39E-01	1.21E+00	6.78E-01	1.39E+00	1.20E+00
	0.1	3.38E-01	9.38E-01	2.03E+00	6.70E-01	1.36E+00	2.00E+00
	0.2	3.30E-01	9.37E-01	3.60E+00	6.91E-01	1.35E+00	3.55E+00
Digits 1, 9 $N = 784$ $n = 2144$	0.05	7.97E-01	2.01E+00	5.61E+00	2.40E+00	3.92E+00	5.58E+00
	0.1	7.91E-01	2.02E+00	9.60E+00	2.36E+00	3.96E+00	9.58E+00
	0.2	8.11E-01	2.02E+00	2.14E+01	2.36E+00	4.39E+00	2.12E+01
Digit1to5 $N = 784$ $n = 3969$	0.05	3.73E+00	5.78E+00	2.03E+01	1.35E+01	1.54E+01	2.11E+01
	0.1	4.07E+00	6.08E+00	4.20E+01	1.21E+01	1.42E+01	4.15E+01
	0.2	4.09E+00	6.11E+00	8.30E+01	1.21E+01	1.40E+01	8.21E+01
Digit6to9 $N = 784$ $n = 5051$	0.05	6.02E+00	8.53E+00	3.54E+01	1.69E+01	2.01E+01	3.39E+01
	0.1	5.96E+00	8.40E+00	7.60E+01	1.74E+01	2.09E+01	7.44E+01
	0.2	5.33E+00	7.73E+00	1.35E+02	1.83E+01	2.18E+01	1.37E+02

6. Conclusion. In this paper, we proposed a convex matrix optimization reformulation of the additive constant problem initially studied in multidimensional scaling and lately used in some of the dimensionality reduction methods. We developed a Newton-CG method and proved its quadratic convergence. We further demonstrate its use in a novel application to the famous LLE method in order to deal with the missing value case. The resulting method is denoted as LLEMDS. The additional computational work in LLEMDS compared to LLE is that it has to compute many local Euclidean distance matrices because of the missing values. This part of computation has to be fast in order for LLEMDS to be competitive. Our numerical results showed that the Newton-CG method is an ideal method for this step. We also demonstrated that even with missing values of considerable high percentages, LLEMDS is still able to reveal the low-dimensional structures hidden in the high-dimensional data.

There are a couple of issues we left out in the application. For example, how to find the “best” neighborhood in LLEMDS? In this paper, we used the K -nearest neighbors based on the distance information available. Another issue is what is the appropriate way to compute the pairwise distances (not necessarily Euclidean) when

there are missing values. Can the geodesic distances be used? Those issues may become essential for some practical problems. Another important application is on visualization of social networks where only certain dissimilarities (one kind of distances) are available among users/agents (see, e.g., [13]). However, the matrix of those dissimilarities is rarely Euclidean. Hence the proposed LLEMDS can be applied to such situations. We hope to explore those topics in future research.

REFERENCES

- [1] S. BAI, H.-D. QI AND N. XIU, *Constrained best Euclidean distance embedding on a sphere: a matrix optimization approach*, SIAM J. Optim., 25 (2015), pp. 439–467.
- [2] J.F. BONNANS AND A. SHAPIRO, *Perturbation Analysis of Optimization Problems*. Springer Series in Operations Research, Springer, 2000.
- [3] I. BORG AND P.J.F. GROENEN, *Modern Multidimensional Scaling: Theory and Applications* (2nd ed.) Springer Series in Statistics, Springer, 2005.
- [4] R. BORSODORF AND N.J. HIGHAM, *A preconditioned Newton algorithm for the nearest correlation matrix*, IMA Journal of Numerical Analysis 94 (2010), pp. 94–107.
- [5] S. CACCIATORE, C. LUCHINAT AND L. TENORI, *Knowledge discovery by accuracy maximization*, Proc. Natl. Acad. Sci. 111 (2014), pp. 5117–5122.
- [6] F. CAILLIEZ, *The analytical solution of the additive constant problem*, Psychometrika 48 (1983), pp. 305–308.
- [7] X. CHEN, H.-D. QI, AND P. TSENG, *Analysis of nonsmooth symmetric matrix valued functions with applications to semidefinite complementarity problems*, SIAM J. Optim. 13 (2003), pp. 960–985.
- [8] Z.X. CHAN AND D.F. SUN, *Constraint nondegeneracy, strong regularity and nonsingularity in semidefinite programming*, SIAM J. Optim. 19, 370–396 (2008).
- [9] H. CHOI AND S. CHOI, *Robust kernel Isomap*, Pattern Recognition 40 (2007), pp. 853–862.
- [10] F.H. CLARKE, *Optimization and Nonsmooth Analysis*, John Wiley & Sons, New York, 1983.
- [11] L.G. COOPER, *A new solution to the additive constant problem in metric multidimensional scaling*, Psychometrika 37 (1972), pp. 311–322.
- [12] T.F. COX AND M.A.A. COX, *Multidimensional Scaling* 2nd Ed, Chapman and Hall/CRC, 2001.
- [13] L.C. FREEMAN, *Graphic techniques for exploring social networks data*, Models and Methods in Social Network Analysis, 2005, pp. 248–269.
- [14] N. GAFFKE AND R. MATHAR, *A cyclic projection algorithm via duality*, Metrika, 36 (1989), pp. 29–54.
- [15] Y. GAO, *Structured Low Rank Matrix Optimization Problems: a Penalty Approach*, PhD Thesis (2010), National University of Singapore.
- [16] J.-B. HIRIART-URRUTY AND C. LEMARÉCHAL, *Convex Analysis and Minimization Algorithms I*, Springer-Verlag, Berlin 1993.
- [17] M.R. HESTENES AND E. STIEFEL, *Methods of conjugate gradients for solving linear systems*, Journal of Research of the National Bureau of Standards 49 (1952), pp. 409–436.
- [18] Y. LECUN, C. CORTES AND C.J.C. BURGES, *Mnist*, 1998. URL: <http://yann.lecun.com/exdb/mnist>.
- [19] J.C. LINGOES, *Some boundary conditions for a monotone analysis of symmetric matrices*, Psychometrika 36 (1971), pp. 195–203.
- [20] C.A. MICCHELLI, *Interpolation of scattered data: distance matrices and conditional positive definite functions*, Constr. Approx. 2 (1986), pp. 11–22.
- [21] H.-D. QI, *A semismooth Newton method for the nearest Euclidean distance matrix problem*, SIAM J. Matrix Anal. Appl. 34 (2013), pp. 67–93.
- [22] H.-D. QI *Conditional quadratic semidefinite programming: examples and methods*, J. Oper. Res. Soc. China 2 (2014), pp. 143170.
- [23] H.-D. QI AND D.F. SUN, *A quadratically convergent Newton method for computing the nearest correlation matrix*, SIAM Journal on Matrix Analysis and Applications 28 (2006), pp. 360–385.
- [24] H.-D. QI AND X.M. YUAN, *Computing the nearest Euclidean distance matrix with low embedding dimensions*, Math. Program. 147 (2014), pp. 351–389.
- [25] L. QI AND J. SUN, *A nonsmooth version of Newton’s method*, Math. Programming 58 (1993), pp. 353–367.
- [26] S.T. ROWEIS AND L.K. SAUL, *Nonlinear dimensionality reduction by locally linear embedding*,

- Science, 290 (2000), pp. 2323–2326.
- [27] L.K. SAUL AND S.T. ROWEIS, *Think globally, fit locally: unsupervised learning of low dimensional manifolds*, J. Machine Learning Res., 4 (2003), pp. 119–155.
- [28] I.J. SCHOENBERG, *Remarks to Maurice Fréchet's article "Sur la définition axiomatique d'une classe d'espaces vectoriels distanciés applicables vectoriellement sur l'espace de Hilbert"*, Ann. Math. 36 (1935), pp. 724–732.
- [29] I.J. SCHOENBERG, *Metric spaces and positive definite functions*, Trans. Amer. Math. Soc., 44 (1938), pp. 522–536.
- [30] F. SHANG, L.C. JIAO, J. SHI AND J. CHAI, *Robust positive semidefinite L-Isomap ensemble*, Pattern Recognition Letters 32 (2011), pp. 640–649.
- [31] H. SIU, L. JIN AND M. XIONG, *Manifold learning for human population structure studies*, PLoS ONE 7 (2012), pp. 1–18.
- [32] D.F. SUN, *The strong second-order sufficient condition and constraint nondegeneracy in nonlinear semidefinite programming and their implications*, Math. Oper. Res. 31, 761–776 (2006)
- [33] D.F. SUN AND J. SUN, *Semismooth matrix valued functions*, Math. Oper. Res. 27 (2002), pp. 150–169.
- [34] J.B. TENENBAUM, V. DE SILVA, AND J.C. LANGFORD, *A global geometric framework for nonlinear dimensionality reduction*, Science, 290 (2000), pp. 2319–2323.
- [35] W.S. TORGERSON, *Multidimensional scaling: I. Theory and method*, Psychometrika, 17 (1952), pp. 401–419.
- [36] J. WANG, *Geometric Structure of High-Dimensional Data and Dimensionality Reduction*, Higher Education Press, Beijing and Springer-Verlag Berlin Heidelberg 2012.
- [37] G. YOUNG AND A.S. HOUSEHOLDER, *Discussion of a set of points in terms of their mutual distances*, Psychometrika 3 (1938), pp. 19–22.
- [38] X. ZHAO, D. SUN AND K.-C. TOH, *A Newton-CG augmented Lagrangian method for semidefinite programming*, SIAM J. Optim. 20 (2010), pp. 1737–1765.