Fast Algorithm for Updating Discriminant Functions in Linear Discriminant Analysis

Kyi Min Htut, Hironori Tamaki, Atsushi Nakajima and Takaomi Shigehara
Department of Information and Computer Sciences,
Faculty of Engineering, Saitama University,
255 Shimo-Okubo, Saitama, Saitama 338-8570, Japan
Tel.: +81-48-858-9035, Fax: +81-48-858-3716
E-mail: sigehara@ics.saitama-u.ac.jp

Abstract: We propose a new numerical algorithm for linear discriminant analysis which makes it possible to *update* the discriminant functions with very low computational cost.

1. Introduction

To clarify the point of the issue, we first give a brief summary for the framework of linear discriminant analysis (LDA). Discriminant analysis is a standard method for statistical pattern recognition [1]. In pattern recognition, we use the term *pattern* to denote the n-dimensional data vector $\boldsymbol{x} \in \mathbf{R}^n$ whose components are measurements of features of an object under consideration. The main purpose in pattern recognition is to determine the *class* $c \in \Gamma \equiv \{1, 2, \cdots, M\}$ to which a given pattern vector $\boldsymbol{x}$ belongs, where M is the number of classes. The problem is reduced to finding the *discriminant functions* $g_c(\boldsymbol{x})$, $c \in \Gamma$, such that

$$g_c(\boldsymbol{x}) = \min_{c' \in \Gamma} g_{c'}(\boldsymbol{x}) \Longrightarrow \boldsymbol{x} \text{ belongs to the class } c. \quad (1)$$

In LDA, the discriminant functions are determined by the following procedure. For given d sample data of pattern $\boldsymbol{x}(t) \in \mathbf{R}^n$ and class $c(t) \in \Gamma$ to which $\boldsymbol{x}(t)$ belongs ($t = 1, 2, \cdots, d$), we first solve the generalized eigenvalue problem (GEP) of n-th order;

$$A_d \boldsymbol{p} = \lambda G_d \boldsymbol{p}, \quad (2)$$

where

$$A_d = \frac{1}{M} \sum_{c=1}^{M} (\bar{\boldsymbol{x}}_d^c - \bar{\boldsymbol{x}}_d)(\bar{\boldsymbol{x}}_d^c - \bar{\boldsymbol{x}}_d)^T \quad (3)$$

and

$$G_d = \frac{1}{d} \sum_{t=1}^{d} \left(\boldsymbol{x}(t) - \bar{\boldsymbol{x}}_d^{c(t)}\right)\left(\boldsymbol{x}(t) - \bar{\boldsymbol{x}}_d^{c(t)}\right)^T \quad (4)$$

are the between-class and within-class variance-covariance matrices, respectively. Here

$$\bar{\boldsymbol{x}}_d = \frac{1}{d} \sum_{t=1}^{d} \boldsymbol{x}(t) \quad \text{and} \quad \bar{\boldsymbol{x}}_d^c = \frac{\sum_{t=1}^{d} \delta(c, c(t)) \boldsymbol{x}(t)}{\sum_{t=1}^{d} \delta(c, c(t))} \quad (5)$$

are the mean vector of the whole sample patterns and the mean vector of the sample patterns which belong to the class c, respectively. (In Eq.(5), $\delta(c, c') = 1$ for $c = c'$ and 0 for $c \neq c'$.) It is important to notice that $M \ll n \ll d$ holds in a generic case; For face identification for example, $M \simeq 10$ is the number of persons to be discriminated, $n \simeq 10^2 \sim 10^3$ is the number of pixels of image and $d \simeq 10^3 \sim 10^4$ is the number of the sample image data. The GEP (2) has at most $M - 1$ nonzero eigenvalues and there indeed exist $M - 1$ nonzero eigenvalues $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_{M-1} > 0$ in a generic case. Let $\boldsymbol{p}_1, \boldsymbol{p}_2, \cdots, \boldsymbol{p}_{M-1}$ be the corresponding normalized eigenvectors; $\boldsymbol{p}_i^T G_d \boldsymbol{p}_j = \delta(i, j)$. Then the discriminant functions in LDA are given by

$$g_c(\boldsymbol{x}) = \left|P^T(\boldsymbol{x} - \bar{\boldsymbol{x}}_d^c)\right|^2 \quad (6)$$

with $P = (\boldsymbol{p}_1 \boldsymbol{p}_2 \cdots \boldsymbol{p}_{M-1})$. Obviously, the most expensive part for determining the discriminant functions (6) is in solving the GEP (2) and it is highly desired to develop an efficient numerical algorithm for the GEP (2). This is exactly our main subject. In this paper, we concentrate on *updating* the discriminant functions when some new sample data are added. In most applications of discriminant analysis, it is very hard to construct the *good* discriminant functions which keep a high rate for correct identification from the sample data given at first and it is necessary to reconstruct the discriminant functions by adding some new sample data. For this purpose, an iterative method based on a nonlinear matrix dynamical system has been proposed in [2]. In this paper, we propose an alternative new numerical algorithm. Our method is based on a direct method and makes it possible to find the *exact* updated discriminant functions only with $O(n^2)$ arithmetic operations, contrary to with $O(n^3)$ arithmetic operations which are required to solve the GEP (2) in a direct manner.

2. Main Results

The standard numerical algorithms for the GEP with a generic constraint such that A_d is symmetric and G_d is positive definite symmetric in Eq.(2) follow the two steps:

(I) Calculate the Cholesky decomposition G_d.

(II) By using the Cholesky decomposition G_d, make the problem reduce to a symmetric eigenvalue problem of n-th order and solve it.

Both of the steps (I) and (II) require $O(n^3)$ arithmetic operations. The first step of our strategy is to make the symmetric eigenvalue problem of n-th order in the step (II) reduce to a mathematically equivalent symmetric eigenvalue problem of M-th order by using a specific feature of the between-class variance-covariance matrix A_d in Eq.(3). Indeed, we can show the following proposition;

Proposition 1 *Assume that the Cholesky decomposition of G_d is given; $G_d = \Lambda_d \Lambda_d^T$ with an n-dimensional lower triangular matrix Λ_d. Then the GEP (2) with Eqs.(3) and (4) reduces to a symmetric eigenvalue problem of M-th order which can be solved with $O(n^2)$ arithmetic operations.*

Proof Let us define an $n \times M$ matrix U_d by

$$U_d = \frac{1}{\sqrt{M}} \left(\bar{\boldsymbol{x}}_d^1 - \bar{\boldsymbol{x}}_d \ \bar{\boldsymbol{x}}_d^2 - \bar{\boldsymbol{x}}_d \ \cdots \ \bar{\boldsymbol{x}}_d^M - \bar{\boldsymbol{x}}_d \right). \qquad (7)$$

Then the between-class variance-covariance matrix can be written as

$$A_d = U_d U_d^T. \qquad (8)$$

By multiplying $U_d^T G_d^{-1} = (\Lambda_d^{-1} U_d)^T \Lambda_d^{-1}$ from the left on both sides, Eq.(2) reduces to an M-th order eigenvalue equation

$$\tilde{A}_d \boldsymbol{q} = \lambda \boldsymbol{q}, \qquad (9)$$

where

$$\boldsymbol{q} = U_d^T \boldsymbol{p} \in \mathbf{R}^M \qquad (10)$$

and the M-dimensional matrix $\tilde{A}_d$ is defined by

$$\tilde{A}_d = \tilde{U}_d^T \tilde{U}_d \qquad (11)$$

with

$$\tilde{U}_d = \Lambda_d^{-1} U_d. \qquad (12)$$

A theorem of singular value decomposition (with a metric matrix G) indicates that the nonzero eigenvalues of Eq.(9) coincide with those of Eq.(2). After finding a nonzero eigenvalue $\lambda \neq 0$ and the associated eigenvector $\boldsymbol{q}$ in Eq.(9), the corresponding eigenvector $\boldsymbol{p}$ in Eq.(2) is obtained by the solution of a linear system

$$G_d \boldsymbol{p} = U_d \boldsymbol{q}. \qquad (13)$$

■

Since $n \gg M$, the reduction of Eq.(2) to Eq.(9) brings about a substantial speed up for numerical computation. If the Cholesky decomposition of G_d is available, we can solve the eigenvalue problem (9) with $O(n^2)$ arithmetic operations. Note that $\tilde{U}_d$ in Eq.(12) can be determined by solving M linear systems with a common n-dimensional lower triangular coefficient matrix; $\Lambda_d \tilde{U}_d = U_d$. Also Eq.(13) is solved with $O(n^2)$ arithmetic operations if the Cholesky decomposition of G_d is completed.

The second step of our strategy is to update the Cholesky decomposition $G_d = \Lambda_d \Lambda_d^T$ with a cheap numerical cost when a new sample data of pattern $\boldsymbol{x}(d+1) \in \mathbf{R}^n$ and class $c(d+1) \in \Gamma$ to which $\boldsymbol{x}(d+1)$ belongs is added. This is indeed possible by the following proposition;

Proposition 2 *Let $G_d = \Lambda_d \Lambda_d^T$ be the Cholesky decomposition of G_d. Assume that a new sample data of pattern $\boldsymbol{x}(d+1) \in \mathbf{R}^n$ and class $c(d+1) \in \Gamma$ is added. Then the Cholesky decomposition of the new within-class variance-covariance matrix*

$$G_{d+1} = \frac{1}{d+1} \sum_{t=1}^{d+1} \left(\boldsymbol{x}(t) - \bar{\boldsymbol{x}}_{d+1}^{c(t)} \right) \left(\boldsymbol{x}(t) - \bar{\boldsymbol{x}}_{d+1}^{c(t)} \right)^T \qquad (14)$$

can be obtained with $O(n^2)$ arithmetic operations.

Proof After a somewhat lengthy but straightforward calculation, we reach

$$\begin{aligned} G_{d+1} &= \frac{d}{d+1} G_d \\ &+ \frac{1}{d+1} \left(\boldsymbol{x}(d+1) - \bar{\boldsymbol{x}}_d^{c(d+1)} \right) \\ &\qquad \times \left(\boldsymbol{x}(d+1) - \bar{\boldsymbol{x}}_d^{c(d+1)} \right)^T \\ &- \frac{\sum_{t=1}^{d+1} \delta(c(d+1), c(t))}{d+1} \left(\bar{\boldsymbol{x}}_{d+1}^{c(d+1)} - \bar{\boldsymbol{x}}_d^{c(d+1)} \right) \\ &\qquad \times \left(\bar{\boldsymbol{x}}_{d+1}^{c(d+1)} - \bar{\boldsymbol{x}}_d^{c(d+1)} \right)^T. \end{aligned} \qquad (15)$$

Eq.(15) shows that G_{d+1} is obtained by adding a rank-one perturbation to G_d and subtracting a rank-one perturbation successively. Thus we can use the technique of the *Cholesky updating/downdating* [3] (see Appendix for details), which makes it possible to obtain the Cholesky decomposition of G_{d+1} from Λ_d with $O(n^2)$ arithmetic operations. ■

Since the updating of the between-class variance-covariance matrix A_d (actually the updating of U_d in Eq.(7)) is carried out with $O(n)$ arithmetic operations, we can update the discriminant functions only with $O(n^2)$ arithmetic operations by combining the methods described in the proof of Propositions 1 and 2.

3. Numerical Experiment

Numerical experiment has been performed on Sun microsystems workstation (OS: Solaris 2.6, CPU: Micro Sparc 100 MHz×2, Main Memory: 128 MB, Compiler: g77 ver.2.95.2 with option '-O3'). In Table 1, we show the execution time for updating the Cholesky decomposition of the within-class variance-covariance matrix G_d when a new sample data of pattern $\boldsymbol{x}(d+1) \in \mathbf{R}^n$ and class $c(d+1) \in \Gamma$ is added. The second line (PM) is the execution time for the proposed method in Proposition 2, while the third line (CM) is the execution time for the conventional method. In the latter case, an additional $O(n^3)$ procedure (matrix product and Cholesky decomposition) is required. We has utilized `dpotrf` routine

Table 1. Execution time for updating Cholesky decomposition of G_d for d = 1000, M = 10. PM and CM are for proposed and conventional methods, respectively.

data dim. n	100	300	500	700	900
PM (sec)	0.01	0.05	0.13	0.28	0.41
CM (sec)	0.94	8.93	25.68	52.87	89.93
speed up	94	178	197	188	219

in LAPACK ver.3.0 for the Cholesky decomposition of the new within-class variance-covariance matrix G_{d+1}. Table 1 shows that the proposed method is quite satisfactory; Compared to the standard $O(n^3)$ procedure, the speed up for updating the within-class variance-covariance matrix G_d by the proposed method is about $100 \sim 200$ for a wide range of the matrix size covering $n \simeq 100 \sim 900$. Together with the method in Proposition 1, we expect to update the discriminant functions within at most a few seconds when a new sample data is added.

One of the authors (T.S.) thanks Dr. Kazuyuki Hiraoka for fruitful discussions and helpful comments.

References

[1] A. Webb, "Statistical Pattern Recognition", Arnold, London, 1999.

[2] K. Hiraoka, M. Hamahira, K. Hidai, H. Mizoguchi, T. Mishima and S. Yoshizawa, "Fast Algorithm for Online Linear Discriminant Analysis", Proceedings of the 2000 International Technical Conference on Circuits/Systems, Computers, and Communications, pp. 274-277, 2000; IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences, Vol. E84-A, pp. 1431-1434, 2001.

[3] G. W. Stewart, "Matrix Algorithms – Volume 1: Basic Decompositions", SIAM, Philadelphia, 1998.

Appendix Cholesky Updating/Downdating

Let G be an $n \times n$ positive definite real symmetric matrix and let

$$G = \Lambda\Lambda^T \tag{16}$$

be the Cholesky decomposition of G, where $\Lambda = (\boldsymbol{l}_1\boldsymbol{l}_2\cdots\boldsymbol{l}_n)$ is a lower triangular matrix. We consider how the Cholesky decomposition (16) of G is perturbed if a rank-one perturbation $\boldsymbol{x}\boldsymbol{x}^T$ $(\boldsymbol{x} \in \mathbf{R}^n)$ is added to G, or subtracted from G. We will see that the new Cholesky decomposition after the rank-one perturbation can be calculated with $O(n^2)$ arithmetic operations by using Λ and $\boldsymbol{x}$ in a direct manner.

A1 Cholesky Updating

We first consider the case that a rank-one perturbation is added to G. Let us define

$$\tilde{G} \equiv G + \boldsymbol{x}\boldsymbol{x}^T. \tag{17}$$

Note that G is symmetric and still positive definite, and hence $\tilde{G}$ has the Cholesky decomposition.

A first note is that Eq.(17) is rewritten as

$$\tilde{G} = \Lambda\Lambda^T + \boldsymbol{x}\boldsymbol{x}^T = \Omega_0\Omega_0^T \tag{18}$$

with an $n \times (n+1)$ matrix

$$\Omega_0 = (\Lambda\ \boldsymbol{x}_0) = (\boldsymbol{l}_1\boldsymbol{l}_2\cdots\boldsymbol{l}_n\boldsymbol{x}_0), \tag{19}$$

where we set $\boldsymbol{x}_0 = \boldsymbol{x}$. So, in order to obtain the Cholesky decomposition of $\tilde{G}$, we have only to find an $(n+1) \times (n+1)$ orthogonal matrix P such that

$$\Omega_0 P = (\Lambda\ \boldsymbol{x}_0)P = (\tilde{\Lambda}\ \mathbf{0}), \tag{20}$$

where $\tilde{\Lambda}$ is a lower triangular matrix, giving the Cholesky decomposition of $\tilde{G}$; $\tilde{G} = \tilde{\Lambda}\tilde{\Lambda}^T$. Note that the matrix Ω_0 in Eq.(19) has such a form as

$$\Omega_0 = \begin{pmatrix} * & & & & & \bullet \\ * & * & & & & \bullet \\ \vdots & \vdots & \ddots & & & \vdots \\ * & * & * & * & & \bullet \\ * & * & * & * & * & \bullet \end{pmatrix}, \tag{21}$$

where asterisks and black dots indicate the non-zero entries. To find an orthogonal matrix P in (20), we first multiply an $(n+1) \times (n+1)$ rotation matrix

$$R_1(\theta_1) = \begin{pmatrix} \cos\theta_1 & & & & -\sin\theta_1 \\ & 1 & & & \\ & & \ddots & & \\ & & & 1 & \\ \sin\theta_1 & & & & \cos\theta_1 \end{pmatrix} \tag{22}$$

with

$$\cos\theta_1 = \frac{l_{1,1}}{\sqrt{l_{1,1}^2 + x_{0,1}^2}}, \quad \sin\theta_1 = \frac{x_{0,1}}{\sqrt{l_{1,1}^2 + x_{0,1}^2}} \tag{23}$$

to Ω_0 from the right. Then we can eliminate the first component (top black dot) in the last column of Ω_0;

$$\begin{aligned}\Omega_1 \equiv \Omega_0 R_1(\theta_1) &= (\tilde{\boldsymbol{l}}_1\boldsymbol{l}_2\cdots\boldsymbol{l}_n\boldsymbol{x}_1) \\ &= \begin{pmatrix} * & & & & & 0 \\ * & * & & & & \bullet \\ \vdots & \vdots & \ddots & & & \vdots \\ * & * & * & * & & \bullet \\ * & * & * & * & * & \bullet \end{pmatrix}\end{aligned} \tag{24}$$

A similar procedure can be applied to eliminate the k-th component in the last column of Ω_0 successively ($k = 2, 3, \cdots, n$). If we define

$$\Omega_k \equiv \Omega_{k-1}R_k(\theta_k) \tag{25}$$

successively by using an $(n+1)\times(n+1)$ rotation matrix

$$R_k(\theta_k)=\begin{array}{cc} & \begin{array}{cccccccc} & & & k & & & & n+1 \end{array} \\ \begin{array}{c} \\ \\ \\ k \\ \\ \\ \\ n+1 \end{array} & \begin{pmatrix} 1 & & & & & & & \\ & \ddots & & & & & & \\ & & 1 & & & & & \\ & & & \cos\theta_k & & & & -\sin\theta_k \\ & & & & 1 & & & \\ & & & & & \ddots & & \\ & & & & & & 1 & \\ & & & \sin\theta_k & & & & \cos\theta_k \end{pmatrix} \end{array} \qquad (26)$$

with

$$\cos\theta_k=\frac{l_{k,k}}{\sqrt{l_{k,k}^2+x_{k-1,k}^2}},\quad \sin\theta_k=\frac{x_{k-1,k}}{\sqrt{l_{k,k}^2+x_{k-1,k}^2}}, \qquad (27)$$

then the matrix Ω_k has such a form as

$$\Omega_k=(\tilde{\boldsymbol{l}}_1\cdots\tilde{\boldsymbol{l}}_k\boldsymbol{l}_{k+1}\cdots\boldsymbol{l}_n\boldsymbol{x}_k), \qquad (28)$$

where the submatrix $(\tilde{\boldsymbol{l}}_1\cdots\tilde{\boldsymbol{l}}_k\boldsymbol{l}_{k+1}\cdots\boldsymbol{l}_n)$ is lower triangular and the first k elements in the last column $\boldsymbol{x}_k$ are zero. By iterating this process for $k=1,2,\cdots,n$, we have

$$\Omega_0P=(\Lambda\ \boldsymbol{x}_0)P=(\tilde{\boldsymbol{l}}_1\tilde{\boldsymbol{l}}_2\cdots\tilde{\boldsymbol{l}}_n\boldsymbol{0}) \qquad (29)$$

with an orthogonal matrix

$$P=R_1(\theta_1)R_2(\theta_2)\cdots R_n(\theta_n). \qquad (30)$$

Since $\tilde{\Lambda}=(\tilde{\boldsymbol{l}}_1\tilde{\boldsymbol{l}}_2\cdots\tilde{\boldsymbol{l}}_n)$ is a lower triangular matrix, it gives the Cholesky decomposition of $\tilde{G}$; $\tilde{G}=\tilde{\Lambda}\tilde{\Lambda}^T$.

A2 Cholesky Downdating

We now turn to the case that a rank-one perturbation is subtracted from G. Let us define

$$\tilde{G}\equiv G-xx^T, \qquad (31)$$

where $\boldsymbol{x}\in\mathbf{R}^n$ is arbitrary as long as $\tilde{G}$ is positive definite. A first note is that Eq.(31) is written as

$$G=\tilde{G}+xx^T=\Omega_0\Omega_0^T \qquad (32)$$

with an $n\times(n+1)$ matrix

$$\Omega_0=(\tilde{\Lambda}\ \boldsymbol{x}_0), \qquad (33)$$

where we set $\boldsymbol{x}_0=\boldsymbol{x}$. Thus, in order to obtain the Cholesky decomposition of $\tilde{G}$, we have only to find an $(n+1)\times(n+1)$ orthogonal matrix P such that

$$\Omega_0P=(\tilde{\Lambda}\ \boldsymbol{x}_0)P=(\Lambda\ \boldsymbol{0}), \qquad (34)$$

where $\tilde{\Lambda}=(\tilde{\boldsymbol{l}}_1\tilde{\boldsymbol{l}}_2\cdots\tilde{\boldsymbol{l}}_n)$ is a lower triangular matrix, giving the Cholesky decomposition of $\tilde{G}$; $\tilde{G}=\tilde{\Lambda}\tilde{\Lambda}^T$. The first step for this purpose is to find a rotation matrix $R_1(\theta_1)$ in Eq.(22) such that

$$(\tilde{\boldsymbol{l}}_1\tilde{\boldsymbol{l}}_2\cdots\tilde{\boldsymbol{l}}_n\boldsymbol{x}_0)R_1(\theta_1)=(\boldsymbol{l}_1\tilde{\boldsymbol{l}}_2\ \cdots\tilde{\boldsymbol{l}}_n\boldsymbol{x}_1), \qquad (35)$$

where the first component of $\boldsymbol{x}_1\in\mathbf{R}^n$ is zero. This is indeed possible as follows. Since $\tilde{l}_{k,1}=0$ ($k=2,3,\cdots,n$) by assumption and $R_1(\theta_1)$ is orthogonal, we have $\tilde{l}_{1,1}^2+x_{0,1}^2=l_{1,1}^2$ or

$$\tilde{l}_{1,1}=\sqrt{l_{1,1}^2-x_{0,1}^2}, \qquad (36)$$

which determines the rotation angle θ_1;

$$\cos\theta_1=\frac{\tilde{l}_{1,1}}{l_{1,1}},\quad \sin\theta_1=\frac{x_{0,1}}{l_{1,1}}. \qquad (37)$$

From the relation

$$\cos\theta_1\tilde{\boldsymbol{l}}_1+\sin\theta_1\boldsymbol{x}_0=\boldsymbol{l}_1, \qquad (38)$$

we have

$$\tilde{\boldsymbol{l}}_1=\frac{\boldsymbol{l}_1-\sin\theta_1\boldsymbol{x}_0}{\cos\theta_1}. \qquad (39)$$

Finally

$$\boldsymbol{x}_1=\cos\theta_1\boldsymbol{x}_0-\sin\theta_1\tilde{\boldsymbol{l}}_1. \qquad (40)$$

A similar procedure can be applied successively. Suppose that we have

$$\Omega_{k-1}=(\boldsymbol{l}_1\cdots\boldsymbol{l}_{k-1}\tilde{\boldsymbol{l}}_k\cdots\tilde{\boldsymbol{l}}_n\boldsymbol{x}_{k-1}), \qquad (41)$$

in advance at the k-th step, where the submatrix $(\boldsymbol{l}_1\cdots\boldsymbol{l}_{k-1}\tilde{\boldsymbol{l}}_k\cdots\tilde{\boldsymbol{l}}_n)$ is lower triangular and the first $k-1$ components of $\boldsymbol{x}_{k-1}$ are zero. Then, using $R_k(\theta_k)$ in Eq.(26), we obtain

$$\Omega_k=\Omega_{k-1}R_k(\theta_k), \qquad (42)$$

namely

$$\begin{aligned}&(\boldsymbol{l}_1\cdots\boldsymbol{l}_{k-1}\boldsymbol{l}_k\tilde{\boldsymbol{l}}_{k+1}\cdots\tilde{\boldsymbol{l}}_n\boldsymbol{x}_k)\\ &\quad=(\boldsymbol{l}_1\cdots\boldsymbol{l}_{k-1}\tilde{\boldsymbol{l}}_k\tilde{\boldsymbol{l}}_{k+1}\cdots\tilde{\boldsymbol{l}}_n\boldsymbol{x}_{k-1})R_k(\theta_k),\end{aligned} \qquad (43)$$

where the rotation angle θ_k, the k-th new vector $\tilde{\boldsymbol{l}}_k$ and $\boldsymbol{x}_k$ are determined as follows;

$$\begin{aligned}\tilde{l}_{k,k}&=\sqrt{l_{k,k}^2-x_{k-1,k}^2},\\ \cos\theta_k=\frac{\tilde{l}_{k,k}}{l_{k,k}},&\qquad \sin\theta_k=\frac{x_{k-1,k}}{l_{k,k}},\\ \tilde{\boldsymbol{l}}_k&=\frac{\boldsymbol{l}_k-\sin\theta_k\boldsymbol{x}_{k-1}}{\cos\theta_k},\\ \boldsymbol{x}_k&=\cos\theta_k\boldsymbol{x}_{k-1}-\sin\theta_k\tilde{\boldsymbol{l}}_k.\end{aligned} \qquad (44)$$

By construction, the first $k-1$ components of $\tilde{\boldsymbol{l}}_k$ as well as the first k components of $\boldsymbol{x}_k$ are all zero. By iterating this process for $k=1,2,\cdots,n$, we have

$$\Omega_0P=(\tilde{\Lambda}\ \boldsymbol{x}_0)P=(\boldsymbol{l}_1\boldsymbol{l}_2\cdots\boldsymbol{l}_n\boldsymbol{0})=(\Lambda\ \boldsymbol{0}) \qquad (45)$$

with an orthogonal matrix

$$P=R_1(\theta_1)R_2(\theta_2)\cdots R_n(\theta_n). \qquad (46)$$

Thus Eq.(34) is established and, in particular, the lower triangular matrix $\tilde{\Lambda}=(\tilde{\boldsymbol{l}}_1\tilde{\boldsymbol{l}}_2\cdots\tilde{\boldsymbol{l}}_n)$ determined by the iterative process in Eq.(44) gives the Cholesky decomposition of $\tilde{G}$; $\tilde{G}=\tilde{\Lambda}\tilde{\Lambda}^T$.