Cholesky Decomposition

Let an $N \times N$ matrix, $\mathbf{A} = [a_{ij}]$, be symmetric, $a_{ij} = a_{ji}$, and positive definite, *i.e.*, $\mathbf{v}^{\mathrm{T}}\mathbf{A}\mathbf{v} > 0$ for any N-element column vector $\mathbf{v}$. Cholesky decomposition constructs a lower triangular matrix, $\mathbf{L} = [l_{ij}]$ ($l_{ij} = 0$ for $i < j$), which "takes the square root of" $\mathbf{A}$:

$$\mathbf{L}\mathbf{L}^{\mathrm{T}} = \mathbf{A} \quad (1)$$

or

$$\sum_{k=1}^{N} l_{ik} l_{jk} = a_{ij} \ (i, j \geq k). \quad (2)$$

For a diagonal element, $i = j$, Eq. (2) reads

$$\sum_{k=1}^{i} l_{ik}^2 = \sum_{k=1}^{i-1} l_{ik}^2 + l_{ii}^2 = a_{ii} \quad (3)$$

or

$$l_{ii} = \sqrt{a_{ii} - \sum_{k=1}^{i-1} l_{ik}^2}. \quad (4)$$

For $j > i$, Eq. (2) reads

$$\sum_{k=1}^{i} l_{ik} l_{jk} = \sum_{k=1}^{i-1} l_{ik} l_{jk} + l_{ii} l_{ji} = a_{ij} \quad (5)$$

or

$$l_{ji} = \frac{1}{l_{ii}}\left(a_{ij} - \sum_{k=1}^{i-1} l_{ik} l_{jk}\right) \ (j = i+1, \dots, N). \quad (6)$$

Equations (4) and (6) constitutes a recursion as follows. First, $l_{11} = \sqrt{a_{11}}$ from Eq. (4) and $l_{j1} = \frac{1}{l_{11}}(a_{12})$ ($j = 2, \dots, N$) from Eq. (6), which determines the first column of $\mathbf{L}$. Next, $l_{22} = \sqrt{a_{22} - l_{21}^2}$ and $l_{j2} = \frac{1}{l_{22}}\left(a_{ij} - l_{21} l_{j1}\right)$ ($j = 3, \dots, N$) to determines the second column. This procedure can be repeated by incrementing column index i at each iteration, since the right-hand sides of Eq. (4) and (6) only contain l_{ji} for lower columns that have already been computed. This can be implemented as the following algorithm.

Algorithm 1: Cholesky decomposition.

for $i = 1{:}N$

  $l_{ii} = \sqrt{a_{ii} - \sum_{k=1}^{i-1} l_{ik}^2}$

  for $j = i{+}1{:}N$

    $l_{ji} = \frac{1}{l_{ii}}\left(a_{ij} - \sum_{k=1}^{i-1} l_{ik} l_{jk}\right)$

Application 1: Orthonormalization

Cholesky decomposition can be used to orthonormalize a basis set of an N-dimensional vector space $\{|\psi_i\rangle | i = 1, \dots, N\}$. Let $\mathbf{S} = \left[s_{ij} = \langle\psi_i|\psi_j\rangle\right]$ be an $N \times N$ overlap matrix. Then, matrix $\mathbf{S}^{\mathrm{T}}\mathbf{S}$ is positive definite and can be Cholesky-decomposed as

$$\mathbf{S}^{\mathrm{T}}\mathbf{S} = \mathbf{L}\mathbf{L}^{\mathrm{T}}. \quad (7)$$

Now consider

$$\mathbf{Q} = \mathbf{S}(\mathbf{L}^{-1})^{\mathrm{T}}, \quad (8)$$

then

$$\mathbf{Q}^{\mathrm{T}}\mathbf{Q} = \mathbf{L}^{-1}\mathbf{S}^{\mathrm{T}}\mathbf{S}(\mathbf{L}^{-1})^{\mathrm{T}} = \mathbf{L}^{-1}\mathbf{L}\mathbf{L}^{\mathrm{T}}(\mathbf{L}^{-1})^{\mathrm{T}} = \mathbf{L}^{-1}\mathbf{L}(\mathbf{L}^{-1}\mathbf{L})^{\mathrm{T}} = \mathbf{I}. \quad (9)$$

Namely, $\mathbf{Q} = \mathbf{S}(\mathbf{L}^{-1})^{\mathrm{T}}$ is orthonormal.

To implement Eq. (8) in a program, let us transpose it as

$$\mathbf{Q}^{\mathrm{T}} = \mathbf{L}^{-1}\mathbf{S}^{\mathrm{T}}. \quad (10)$$

By denoting the i-th row vectors of $\mathbf{Q}$ and $\mathbf{S}$ as $\mathbf{q}_i$ and $\mathbf{s}_i$, respectively,

$$\mathbf{q}_i = \mathbf{L}^{-1}\,\mathbf{s}_i \ (i = 1, \dots, N), \quad (11)$$

which amounts to solving a linear system of equations,

$$\mathbf{L}\mathbf{q}_i = \mathbf{s}_i \ (i = 1, \dots, N). \quad (12)$$

The lower triangular linear system, Eq. (12), can be solved by recursion. By dropping the row-vector index for simplicity as, $\mathbf{Lq} = \mathbf{s}$, the recursion reads:

$$q_1 = \frac{s_1}{l_{11}}$$
$$q_i = \frac{1}{l_{ii}}\left(s_i - \sum_{j=1}^{i-1} l_{ij}q_j\right) \ (i = 2, \dots, N) \quad (13)$$

Application 2: Low-Rank Approximation

Let us rewrite Cholesky decomposition in Eq. (2) as

$$a_{ij} = \sum_{k=1}^{\min(i,j)} l_{ik}l_{jk}. \quad (14)$$

A low-rank approximation of matrix $\mathbf{A}$ can be obtained by truncating the k-sum in Eq. (14) at $k \le m \ll N$. This is achieved by swapping rows and columns at each Cholesky iteration so that the largest diagonal element is placed at the top of the currently considered submatrix [*cf.* G. H. Golub and C. F. van Loan, *Matrix Computation, 2nd Ed*. (Johns Hopkins Univ. Press, 1989) Sec. 4.2.9]. This is implemented in the following pivoted Cholesky algorithm and truncating the iteration when the largest remaining diagonal element falls below a prescribed threshold δ. Upon the termination of the algorithm, m is the rank of the approximation and the resulting rank-m approximation of $\mathbf{A}$ is given by

$$a_{ij} \cong \sum_{k=1}^{\min(i,j,m)} l_{ik}l_{jk}. \quad (15)$$

Algorithm 2: Pivoted Cholesky decomposition.

for $i = 1{:}N$
    $q = \underset{k\in[i,N]}{\operatorname{argmax}}\, a_{kk}$
    if $a_{qq} < \delta$
        $m = i - 1$
        break
    $a_{i,:} \leftrightarrow a_{q,:}$ // Swap i-th and m-th rows
    $a_{:,i} \leftrightarrow a_{:,q}$ // Swap i-th and m-th columns
    $l_{ii} = \sqrt{a_{ii} - \sum_{k=1}^{i-1} l_{ik}^2}$
    for $j = i{+}1{:}N$
        $l_{ji} = \frac{1}{l_{ii}}\left(a_{ij} - \sum_{k=1}^{i-1} l_{ik}l_{jk}\right)$

Numerical Recipes Program for Cholesky Decomposition

Source Codes

- Cholesky decomposition: https://aiichironakano.github.io/phys516/src/TB/choldc.c
- Driver: https://aiichironakano.github.io/phys516/src/TB/cholesky.c

Compile and Run

```
$ cc -o cholesky cholesky.c choldc.c -lm
$ ./cholesky
```

```
A
1.000000e+00 2.000000e-01 1.000000e-01
2.000000e-01 1.000000e+00 3.000000e-01
1.000000e-01 3.000000e-01 1.000000e+00

L
1.000000e+00
2.000000e-01 9.797959e-01
1.000000e-01 2.857738e-01 9.530652e-01

L•Lt
1.000000e+00 2.000000e-01 1.000000e-01
2.000000e-01 1.000000e+00 3.000000e-01
1.000000e-01 3.000000e-01 1.000000e+00
```

Numerical Recipes Section 2.9: Cholesky Decomposition

https://aiichironakano.github.io/phys516/c2-9.pdf