5. The Cholesky factorization

- positive (semi-)definite matrices
- examples
- the Cholesky factorization
- solving $Ax = b$ with $A$ positive definite
- inverse of a positive definite matrix
- permutation matrices
- sparse Cholesky factorization

Positive (semi-)definite matrices

- $A$ is positive definite if $A$ is symmetric and
  \[ x^T A x > 0 \text{ for all } x \neq 0 \]
- $A$ is positive semidefinite if $A$ is symmetric and
  \[ x^T A x \geq 0 \text{ for all } x \]

Note: if $A$ is symmetric of order $n$, then

\[ x^T A x = \sum_{i=1}^{n} \sum_{j=1}^{n} a_{ij} x_i x_j = \sum_{i=1}^{n} a_{ii} x_i^2 + 2 \sum_{i>j} a_{ij} x_i x_j \]

Examples

\[ A_1 = \begin{bmatrix} 9 & 6 \\ 6 & 5 \end{bmatrix}, \quad A_2 = \begin{bmatrix} 9 & 6 \\ 6 & 4 \end{bmatrix}, \quad A_3 = \begin{bmatrix} 9 & 6 \\ 6 & 3 \end{bmatrix} \]

- $A_1$ is positive definite:
  \[ x^T A_1 x = 9x_1^2 + 12x_1x_2 + 5x_2^2 = (3x_1 + 2x_2)^2 + x_2^2 \]
- $A_2$ is positive semidefinite but not positive definite:
  \[ x^T A_2 x = 9x_1^2 + 12x_1x_2 + 4x_2^2 = (3x_1 + 2x_2)^2 \]
- $A_3$ is not positive semidefinite:
  \[ x^T A_3 x = 9x_1^2 + 12x_1x_2 + 3x_2^2 = (3x_1 + 2x_2)^2 - x_2^2 \]

Examples

- $A = B^T B$ for some matrix $B$
  \[ x^T A x = x^T B^T B x = \| B x \|^2 \]
  $A$ is positive semidefinite
  $A$ is positive definite if $B$ has a zero nullspace
- diagonal $A$
  \[ x^T A x = a_{11} x_1^2 + a_{22} x_2^2 + \cdots + a_{nn} x_n^2 \]
  $A$ is positive semidefinite if its diagonal elements are nonnegative
  $A$ is positive definite if its diagonal elements are positive
Another example

\[
A = \begin{bmatrix}
1 & -1 & \cdots & 0 & 0 \\
-1 & 2 & \cdots & 0 & 0 \\
\vdots & \vdots & \ddots & \vdots & \vdots \\
0 & 0 & \cdots & 2 & -1 \\
0 & 0 & \cdots & -1 & 1 \\
\end{bmatrix}
\]

\(A\) is positive semidefinite:

\[
x^T A x = (x_1 - x_2)^2 + (x_2 - x_3)^2 + \cdots + (x_{n-1} - x_n)^2 \geq 0
\]

\(A\) is not positive definite:

\[
x^T A x = 0 \text{ for } x = (1, 1, \ldots, 1)
\]

Resistor circuit

Circuit model: \(y = Ax\) with

\[
A = \begin{bmatrix}
R_1 + R_3 & R_3 \\
R_3 & R_2 + R_3 \\
\end{bmatrix} \quad (R_1, R_2, R_3 > 0)
\]

Interpretation of \(x^T A x = y^T x\)

\(x^T A x\) is the power delivered by the sources, dissipated by the resistors

Properties

if \(A\) is positive definite of order \(n\), then

- \(A\) has a zero nullspace
  
  \[
  \text{proof: } x^T A x > 0 \text{ for all nonzero } x, \text{ hence } Ax \neq 0 \text{ if } x \neq 0
  \]

- the diagonal elements of \(A\) are positive
  
  \[
  \text{proof: } a_{ii} = e_i^T A e_i > 0 \quad (e_i \text{ is the } i\text{th unit vector})
  \]

- \(A_{22} - (1/a_{11})A_{21}A_{21}^T\) is positive definite, where \(A = \begin{bmatrix}
a_{11} & A_{21}^T \\
A_{21} & A_{22} \\
\end{bmatrix}\)

  \[
  \text{proof: } \text{take any } v \neq 0 \text{ and } w = -(1/a_{11})A_{21}^T v
  \]

  \[
  v^T \left( A_{22} - \frac{1}{a_{11}} A_{21}A_{21}^T \right) v = \begin{bmatrix} w \ v^T \end{bmatrix} \begin{bmatrix}
a_{11} & A_{21}^T \\
A_{21} & A_{22} \\
\end{bmatrix} \begin{bmatrix} w \\ v \end{bmatrix} > 0
  \]
**Cholesky factorization**

every positive definite matrix \( A \) can be factored as

\[
A = LL^T
\]

where \( L \) is lower triangular with positive diagonal elements

**Cost:** \((1/3)n^3\) flops if \( A \) is of order \( n \)

- \( L \) is called the **Cholesky factor** of \( A \)
- can be interpreted as 'square root' of a positive definite matrix

**Proof** that the algorithm works for positive definite \( A \) of order \( n \)

- step 1: if \( A \) is positive definite then \( a_{11} > 0 \)
- step 2: if \( A \) is positive definite, then

\[
A_{22} - L_{21}L_{21}^T = A_{22} - \frac{1}{a_{11}} A_{21}A_{21}^T
\]

is positive definite (see page 5–8)

- hence the algorithm works for \( n = m \) if it works for \( n = m - 1 \)
- it obviously works for \( n = 1 \); therefore it works for all \( n \)

---

**Example**

\[
\begin{bmatrix}
25 & 15 & -5 \\
15 & 18 & 0 \\
-5 & 0 & 11
\end{bmatrix} = \begin{bmatrix}
l_{11} & 0 & 0 \\
l_{21} & l_{22} & 0 \\
l_{31} & l_{32} & l_{33}
\end{bmatrix} \begin{bmatrix}
l_{11} & l_{21} & l_{31} \\
l_{21} & l_{22} & l_{32} \\
l_{31} & l_{32} & l_{33}
\end{bmatrix}
\]

- first column of \( L \)

\[
\begin{bmatrix}
25 & 15 & -5 \\
15 & 18 & 0 \\
-5 & 0 & 11
\end{bmatrix} = \begin{bmatrix}
5 & 0 & 0 \\
3 & l_{22} & 0 \\
-1 & l_{32} & l_{33}
\end{bmatrix} \begin{bmatrix}
5 & 3 & -1 \\
0 & l_{22} & l_{32} \\
0 & 0 & l_{33}
\end{bmatrix}
\]

- second column of \( L \)

\[
\begin{bmatrix}
18 & 0 \\
0 & 11
\end{bmatrix} - \begin{bmatrix}
3 \\
-1
\end{bmatrix} \begin{bmatrix}
l_{22} & 0 \\
l_{32} & l_{33}
\end{bmatrix} = \begin{bmatrix}
l_{22} & 0 \\
l_{32} & l_{33}
\end{bmatrix} \begin{bmatrix}
l_{22} & l_{32} \\
l_{32} & l_{33}
\end{bmatrix}
\]

\[
\begin{bmatrix}
9 & 3 \\
3 & 10
\end{bmatrix} = \begin{bmatrix}
3 & 0 \\
1 & l_{33}
\end{bmatrix} \begin{bmatrix}
3 & 1 \\
0 & l_{33}
\end{bmatrix}
\]

---

**Cholesky factorization algorithm**

partition matrices in \( A = LL^T \) as

\[
\begin{bmatrix}
a_{11} & A_{21}^T \\
A_{21} & A_{22}
\end{bmatrix} = \begin{bmatrix}
l_{11} & 0 \\
L_{21} & L_{22}
\end{bmatrix} \begin{bmatrix}
l_{11} & L_{21}^T \\
L_{21} & L_{22}
\end{bmatrix}
\]

\[
= \begin{bmatrix}
l_{11}^2 & l_{11}L_{21}^T \\
l_{11}L_{21} & l_{11}L_{21} + L_{22}
\end{bmatrix}
\]

**Algorithm**

1. determine \( l_{11} \) and \( L_{21} \): \( l_{11} = \sqrt{a_{11}}, \quad L_{21} = \frac{1}{l_{11}} A_{21} \)

2. compute \( L_{22} \) from

\[
A_{22} - L_{21}L_{21}^T = L_{22}L_{22}^T
\]

this is a Cholesky factorization of order \( n - 1 \)
• third column of $L$: $10 - 1 = l_{33}^2$, i.e., $l_{33} = 3$

**Conclusion:**

\[
\begin{bmatrix}
25 & 15 & -5 \\
15 & 18 & 0 \\
-5 & 0 & 11
\end{bmatrix} =
\begin{bmatrix}
5 & 0 & 0 \\
3 & 3 & 0 \\
-1 & 1 & 3
\end{bmatrix}
\begin{bmatrix}
5 & 3 & -1 \\
0 & 3 & 1 \\
0 & 0 & 3
\end{bmatrix}
\]

---

**Solving equations with positive definite $A$**

$Ax = b$ \hspace{1em} (A positive definite of order $n$)

**Algorithm**

- factor $A$ as $A = LL^T$
- solve $LL^Tx = b$
  - forward substitution $Lz = b$
  - back substitution $L^Tx = z$

**Cost:** $(1/3)n^3$ flops

- factorization: $(1/3)n^3$
- forward substitution: $n^2$
- backward substitution: $n^2$

---

**Inverse of a positive definite matrix**

suppose $A$ is positive definite with Cholesky factorization $A = LL^T$

- $L$ is invertible (its diagonal is nonzero; see lecture 4)
- $X = L^{-T}L^{-1}$ is a right inverse of $A$:
  \[
  AX = LL^T L^{-T} L^{-1} = LL^{-1} = I
  \]
- $X = L^{-T}L^{-1}$ is a left inverse of $A$:
  \[
  XA = L^{-T}L^{-1} LL^T = L^{-T}L^T = I
  \]
- hence, $A$ is invertible and
  \[
  A^{-1} = L^{-T}L^{-1}
  \]

---

**Summary**

if $A$ is positive definite of order $n$

- $A$ can be factored as $LL^T$
- the cost of the factorization is $(1/3)n^3$ flops
- $Ax = b$ can be solved in $(1/3)n^3$ flops
- $A$ is invertible: $A^{-1} = L^{-T}L^{-1}$
- $A$ has a full range: $Ax = b$ is solvable for all $b$
- $A$ has a zero nullspace: $x^TAx > 0$ for all nonzero $x$
Sparse positive definite matrices

- a matrix is \textit{sparse} if most of its elements are zero
- a matrix is \textit{dense} if it is not sparse

Cholesky factorization of dense matrices

- \((1/3)n^3\) flops
- on a current PC: a few seconds or less, for \(n\) up to a few 1000

Cholesky factorization of sparse matrices

- if \(A\) is very sparse, then \(L\) is often (but not always) sparse
- if \(L\) is sparse, the cost of the factorization is much less than \((1/3)n^3\)
- exact cost depends on \(n\), \#nonzero elements, sparsity pattern
- very large sets of equations \((n \sim 10^6)\) are solved by exploiting sparsity

Permutation matrices

- a permutation matrix is the identity matrix with its rows reordered, e.g.,
  \[
  \begin{bmatrix}
  0 & 1 & 0 \\
  1 & 0 & 0 \\
  0 & 0 & 1 \\
  \end{bmatrix},
  \begin{bmatrix}
  0 & 1 & 0 \\
  0 & 0 & 1 \\
  1 & 0 & 0 \\
  \end{bmatrix}
  \]
- the vector \(Ax\) is a permutation of \(x\)
  \[
  \begin{bmatrix}
  0 & 1 & 0 \\
  0 & 0 & 1 \\
  1 & 0 & 0 \\
  \end{bmatrix}
  \begin{bmatrix}
  x_1 \\
  x_2 \\
  x_3 \\
  \end{bmatrix}
  =
  \begin{bmatrix}
  x_2 \\
  x_3 \\
  x_1 \\
  \end{bmatrix}
  \]
- \(A^T x\) is the inverse permutation applied to \(x\)
  \[
  \begin{bmatrix}
  0 & 0 & 1 \\
  1 & 0 & 0 \\
  0 & 1 & 0 \\
  \end{bmatrix}
  \begin{bmatrix}
  x_1 \\
  x_2 \\
  x_3 \\
  \end{bmatrix}
  =
  \begin{bmatrix}
  x_3 \\
  x_1 \\
  x_2 \\
  \end{bmatrix}
  \]
- \(A^T A = AA^T = I\), so \(A\) is invertible and \(A^{-1} = A^T\)
Solving \( Ax = b \) when \( A \) is a permutation matrix

The solution of \( Ax = b \) is \( x = A^Tb \)

Example

\[
\begin{bmatrix}
0 & 1 & 0 \\
0 & 0 & 1 \\
1 & 0 & 0
\end{bmatrix}
\begin{bmatrix}
x_1 \\
x_2 \\
x_3
\end{bmatrix} =
\begin{bmatrix}
1.5 \\
10.0 \\
-2.1
\end{bmatrix}
\]

Solution is \( x = (-2.1, 1.5, 10.0) \)

Cost: zero flops

Sparse Cholesky factorization

If \( A \) is sparse and positive definite, it is usually factored as

\[
A = PLL^T P^T
\]

\( P \) a permutation matrix; \( L \) lower triangular with positive diagonal elements

Interpretation: we permute the rows and columns of \( A \) and factor

\[
P^T AP = LL^T
\]

- choice of \( P \) greatly affects the sparsity \( L \)
- many heuristic methods (that we don’t cover) exist for selecting good permutation matrices \( P \)

Solving sparse positive definite equations

Solve \( Ax = b \) via factorization \( A = PLL^T P^T \)

Algorithm

1. \( \tilde{b} := P^Tb \)
2. solve \( Lz = \tilde{b} \) by forward substitution
3. solve \( L^Ty = z \) by back substitution
4. \( x := Py \)

Interpretation: we solve

\[
(P^T AP)y = \tilde{b}
\]

using the Cholesky factorization of \( P^T AP \)