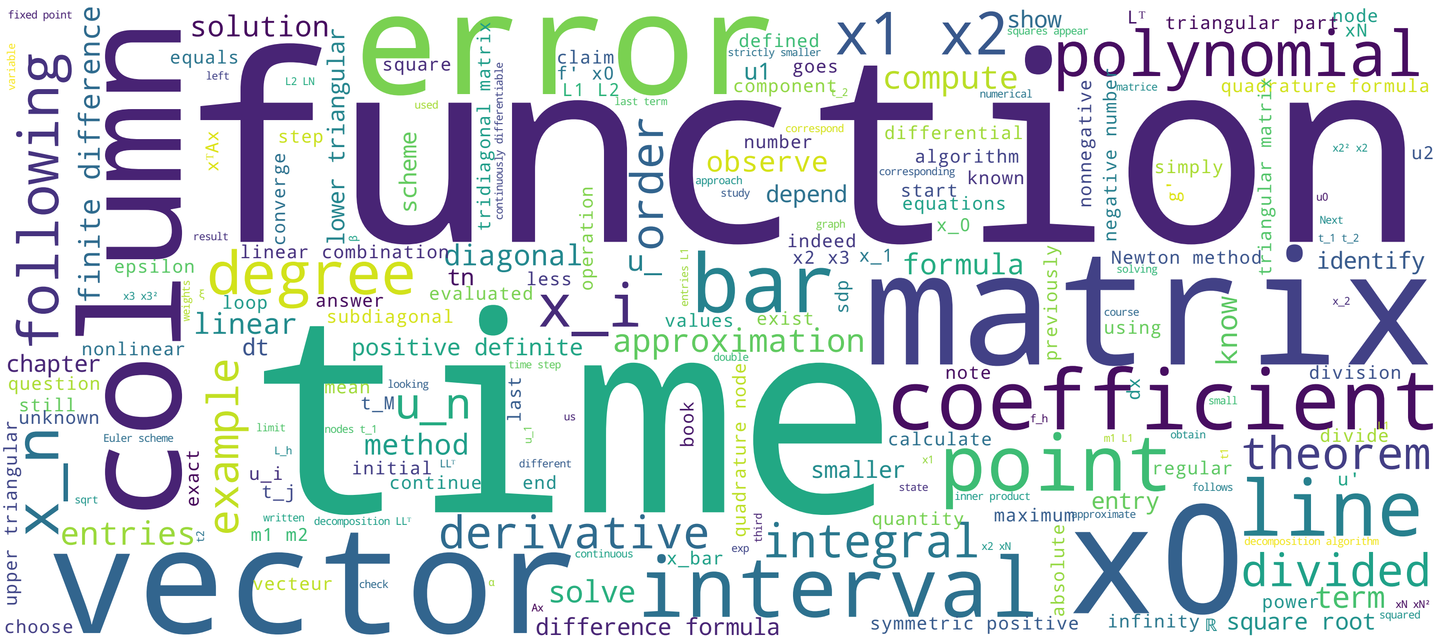

Chap 5 : Décomposition LL^T - Exemple


function
time
matrix
column
error
vector
polynomial
coefficient
x0
interval
point
degree
following
finite difference
derivative
integral
line
divided
bar
x_i
x_n
approximation
diagonal
example
solution
x1 x2
order
theorem
formula
compute
observe
depend
differential
algorithm
positive definite
linear combination
lower triangular
tridiagonal matrix
scheme
method
u_n
entries
solve
difference formula
symmetric positive
square root
Newton method
identify
quadrature formula
triangular part
nonnegative number
upper triangular
chapter
question
unknown
nonlinear
choose

Video

EPFL

Chap 5 : Décomposition LL^T - Exemple

$$\underbrace{\begin{pmatrix} 2 & -1 & & & \\ -1 & 2 & -1 & & \\ & -1 & \ddots & \ddots & \\ & & \ddots & 2 & -1 \\ & & & -1 & 2 \end{pmatrix}}_{A} = \underbrace{\begin{pmatrix} l_1 & & & (0) & \\ m_1 & l_2 & & & \\ & m_2 & \ddots & & \\ & & \ddots & \ddots & \\ & & & m_{N-1} & l_N \end{pmatrix}}_{L} \underbrace{\begin{pmatrix} l_1 & m_1 & & & \\ & l_2 & m_2 & & \\ & & \ddots & \ddots & \\ & (0) & & l_{N-1} & m_{N-1} \\ & & & & l_N \end{pmatrix}}_{L^T}$$

$$\forall \vec{x} \in \mathbb{R}^N \quad \vec{x}^T A \vec{x} = (x_1 \; x_2 \; \cdots \; x_N) \begin{pmatrix} 2 & -1 & & \\ -1 & \ddots & \ddots & \\ & \ddots & \ddots & -1 \\ & & -1 & 2 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_N \end{pmatrix}$$

Let's study the decomposition LL^T with the example of a tridiagonal matrix. So the matrix A is the tridiagonal matrix with 2's on the main diagonal, -1 on the diagonal subdiagonal and -1 on the diagonal superdiagonal. We will first show that this matrix A is symmetric and positive definite, we know that ithere is a decomposition LL^T, where L is a lower triangular matrix, and consequently L^T is an upper matrix, moreover, the coefficients on the diagonal of L are stricly positive. I will note those entries L1, L2, until LN and so for L^T I will also have L1, L2,..., LN. The matrix L is lower triangular, so there are 0's in the upper triangular part, and so here 0's on the lower triangular part of L^T. Actually, the lower part of L reduces to a subdiagonal. The subdiagonal of L has coefficients m1, m2, until m(N-1) and so L^T is reduced to a superdiagonal m1, m2, until m(N-1). Here is L(N-1). First of all, let's show that this matrix A is indeed symmetric positive definite. We take any x in $\mathbb{R}^N$ and compute $x^T Ax$, So x^T is the vector with entries x1, x2, until xN, laying down. We have the matrix A, with 2 on the diagonal, -1 on the subdiagonal, -1 on the superdiagonal, and the vector x, x1, x2,..., xN.

Notes

Summary

0m 00s

Chap 5 : Décomposition LL^T - Exemple

$$\underbrace{\begin{pmatrix} 2 & -1 & & & \\ -1 & 2 & -1 & & \\ & -1 & \ddots & \ddots & \\ & & \ddots & 2 & -1 \\ & & & -1 & 2 \end{pmatrix}}_{A} = \underbrace{\begin{pmatrix} l_1 & & & (0) \\ m_1 & l_2 & & \\ & m_2 & \ddots & \\ & & \ddots & \ddots & \\ & & & m_{N-1} & l_N \end{pmatrix}}_{L} \underbrace{\begin{pmatrix} l_1 & m_1 & & & \\ & l_2 & m_2 & & \\ & & \ddots & \ddots & \\ & (0) & & l_{N-1} & m_{N-1} \\ & & & & l_N \end{pmatrix}}_{L^T}$$

$$\forall \vec{x} \in \mathbb{R}^N \quad \vec{x}^T A \vec{x} = (x_1 \ x_2 \ \cdots \ x_N) \begin{pmatrix} 2 & -1 & & \\ -1 & \ddots & \ddots & \\ & \ddots & \ddots & -1 \\ & & -1 & 2 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_N \end{pmatrix} = 2x_1^2 - 2x_1x_2 + 2x_2^2 - 2x_2x_3 + \cdots - 2x_{N-1}x_N + 2x_N^2$$

$$= x_1^2 + (x_1^2 - 2x_1x_2 + x_2^2) + (x_2^2 - 2x_2x_3 + x_3^2) + \cdots + (x_{N-1}^2 - 2x_{N-1}x_N + x_N^2) + x_N^2$$

$$= x_1^2 + (x_1 - x_2)^2 + (x_2 - x_3)^2 + \cdots + (x_{N-1} - x_N)^2 + x_N^2$$

If I do the computation of $x^T Ax$ I obtain $2*x1^2$ plus $2*x2^2$ donc il y a *2x1²*, then, here I have $-x1*x2$, donc j'ai - *x1 x2*, et en fait, j'ai $-2*x1*x2+2*x2^2$, $-2*x2*x3$, $+2*x3^2$, until the end, where there is $-2*x(N-1)*xN$, and the last term will be $2xN^2$. I'd like to show that $x^T Ax$ is nonnegative, so a possibility is to make squares appear, $2x1^2$ is $x1^2+x1^2$, We have the double product $-2*x1*x2$, and I have $x2^2$, so I can keep this term, I observe a square, then I'll have $x2^2-2*x2*x3+x3^2$, and so on until the last term which will be here $x(N-1)^2+2*x(N-1)xN+xN^2$, and again xN^2. So observe now that squares appear, $x1^2+(x1-x2)^2 +(x2-x3)^2 +$... etc... $(x(N-1)-xN)^2+xN^2$. So $x^T Ax$ is indeed a sum of squares, so this term is nonnegative. Moreover, $x^T Ax =0$ if and only if all those terms are 0, since they are squares. So I'll have x1=0, if x1=0 I'll get x2=0, and so on until I get xN=0. Obviously this matrix is symmetric, so I've shown that this matrix A is sdp. Let's move to the Cholesky decomposition algorithm. I told you that we have to identify the coefficients of A and LL^T in the right order, so I will start with the first column.

Notes

Summary

2m 13s

Chap 5 : Décomposition LL^T - Exemple

$$\underset{A}{\begin{pmatrix} 2 & -1 & & & \\ -1 & 2 & -1 & & \\ & \ddots & \ddots & \ddots & \\ & & & 2 & -1 \\ & & & -1 & 2 \end{pmatrix}} = \underset{L}{\begin{pmatrix} l_1 & & & (0) \\ m_1 & l_2 & & \\ & m_2 & \ddots & \\ & & \ddots & \ddots & \\ & & & m_{N-1} & l_N \end{pmatrix}} \underset{L^T}{\begin{pmatrix} l_1 & m_1 & & \\ & l_2 & m_2 & \\ & & \ddots & \ddots \\ & (0) & & l_{N-1} & m_{N-1} \\ & & & & l_N \end{pmatrix}}$$

$$\forall \vec{x} \in \mathbb{R}^N \quad \vec{x}^T A \vec{x} = (x_1\ x_2\ \cdots\ x_N) \begin{pmatrix} 2 & -1 & & \\ -1 & \ddots & \ddots & \\ & \ddots & \ddots & -1 \\ & & -1 & 2 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_N \end{pmatrix} = 2x_1^2 - 2x_1x_2 + 2x_2^2 - 2x_2x_3 + \cdots - 2x_{N-1}x_N + 2x_N^2$$

$$= x_1^2 + (x_1^2 - 2x_1x_2 + x_2^2) + (x_2^2 - 2x_2x_3 + x_3^2) + \cdots + (x_{N-1}^2 - 2x_{N-1}x_N + x_N^2) + x_N^2$$

$$= x_1^2 + (x_1 - x_2)^2 + (x_2 - x_3)^2 + \cdots + (x_{N-1} - x_N)^2 + x_N^2$$

$2 = l_1^2 \rightarrow l_1 = \sqrt{2}$

$-1 = m_1 l_1 \rightarrow m_1 = -1/l_1$

$2 = m_1^2 + l_2^2 \rightarrow l_2 = \sqrt{2 - m_1^2}$

Algo : $\vec{l}, \vec{m}, \cancel{\vec{u}}$

$l_1 = \sqrt{2}$

Faire $i = 1, N-1$

$m_i = -1/l_i$

The entries of the first column of A will give me the coefficients of the first column of L, indeed the coefficient 2 here is the entry 1,1 of the matrix A, I must do the inner produt between the first line and the first column, so I get 2=L1², so L1=√2. I continue, I get the coefficients of the second line and first column, -1 equals the second line times the first column, m1*L1. I get m1=-1/L1. And so on I identify the coefficiens here of the second column of A, with the coefficients of the second column of the product LLᵀ, and I'll get the coefficients of the second column of L. I continue, I have 2, the entry 2,2 of the matrix A which equals the inner product of this line with this column, which is m1²+L2², I get L2 = √(2 - m1²), and so on. Now I can write the LLᵀ decomposition algorithm. « Algorithm » This algorithm uses the vector L, the vector of entries L1, L2,..., LN, the vector m, which is the vector with entries m1, m2 until m(N-1). et le vecteur ***u***, qui est -- et c'est tout *(**u** barré)*. So I initialize the algorithm, the initialization is L1=√2, and then I do a loop for i from 1 to N-1, From L1, I will compute m1, from Li I will compute mi, which equals -1/Li.

Notes

Summary

5m 01s

Chap 5 : Décomposition LL^T - Exemple

$$\underset{A}{\begin{pmatrix} \boxed{\begin{matrix}2\\-1\end{matrix}} & -1 & & \\ & \boxed{\begin{matrix}2\\-1\end{matrix}} & -1 & \\ & \ddots & \ddots & \ddots \\ & & -1 & 2 \end{pmatrix}} = \underset{L}{\begin{pmatrix} \boxed{\begin{matrix}l_1\\m_1\end{matrix}} & & (0) & \\ & \boxed{\begin{matrix}l_2\\m_2\end{matrix}} & & \\ & \ddots & \ddots & \\ & & m_{N-1} & l_N \end{pmatrix}} \underset{L^T}{\begin{pmatrix} l_1 & m_1 & & \\ & l_2 & m_2 & \\ (0) & & \ddots & \ddots \\ & & & l_{N-1} \; m_{N-1} \\ & & & l_N \end{pmatrix}}$$

$$\forall \vec{x} \in \mathbb{R}^N \quad \vec{x}^T A \vec{x} = (x_1 \; x_2 \; \cdots \; x_N) \begin{pmatrix} 2 & -1 & & \\ -1 & \ddots & \ddots & \\ & \ddots & \ddots & -1 \\ & & -1 & 2 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_N \end{pmatrix} = 2x_1^2 - 2x_1x_2 + 2x_2^2 - 2x_2x_3 + \cdots - 2x_{N-1}x_N + 2x_N^2$$

$$= x_1^2 + (x_1^2 - 2x_1x_2 + x_2^2) + (x_2^2 - 2x_2x_3 + x_3^2) + \cdots + (x_{N-1}^2 - 2x_{N-1}x_N + x_N^2) + x_N^2$$

$$= x_1^2 + (x_1 - x_2)^2 + (x_2 - x_3)^2 + \cdots + (x_{N-1} - x_N)^2 + x_N^2$$

$2 = l_1^2 \rightarrow l_1 = \sqrt{2}$

$-1 = m_1 l_1 \rightarrow m_1 = -1/l_1$

$2 = m_1^2 + l_2^2 \rightarrow l_2 = \sqrt{2 - m_1^2}$

Algo : $\vec{l}, \vec{m}$, $\cancel{\vec{x}}$

$l_1 = \sqrt{2}$

Faire $i = 1, N-1$

$m_i = -1/l_i$

$l_{i+1} = \sqrt{2 - (m_i)^2}$

Division par zéro, racine carrée nbre négatif ?

Non

Nombre d'opération $O(N)$

Then I am able to compute l(i+1) which equals √(2 - mi)². That is our algorithm, as previously, we can check several things. The first one is: is there is division by 0 or a negative square root? - the square root of a negative number. The answer is no. Why? Since the matrix is sdp, it is regular, and what I did not mention is, that if this this matrix is symmetric positive definite, then all the principal submatrix are also regular. Finally if you get the square root of a negative number, then by contraposition, then the matrix A isn't symmetric positive definite. So there are no divisions by 0 and no square root of negative numbers. The number of operations is still O(N), which means it's doubled each time we double N, simply because there is only one loop from 1 to N.

Notes

Summary

7m 52s