Special Matrices

Sparse Matrix: A matrix is sparse if large fraction of its entries are zero ← No concrete def^n of sparsity, in general

Examples: Band matrices

- lower bandwidth: $A \in \mathbb{R}^{m \times n}$ has lower bandwidth p if $a_{ij} = 0$ whenever $i > j + p$
- upper bandwidth: $A \in \mathbb{R}^{m \times n}$ has upper bandwidth q if $a_{ij} = 0$ whenever $j > i + q$

Quiz: Find the bandwidth of following matrix:

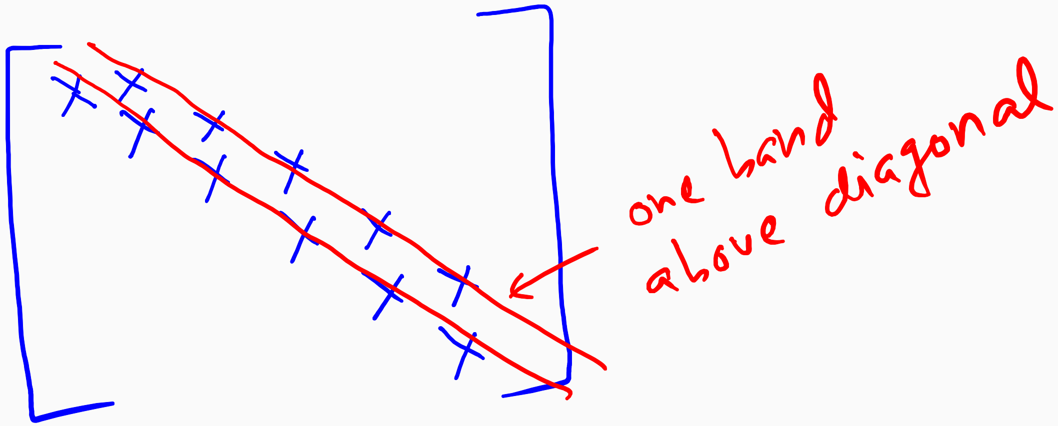
$$\begin{pmatrix} x & x & x & 0 & 0 \\ x & x & x & x & 0 \\ 0 & x & x & x & x \\ 0 & 0 & x & x & x \\ 0 & 0 & 0 & x & x \\ 0 & 0 & 0 & 0 & x \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

1. What is lower bandwidth? $\underline{1}$
2. What is upper bandwidth? $\underline{2}$

Upper Bidiagonal:

low. bandwidth = 0, upp. bandwidth = 1

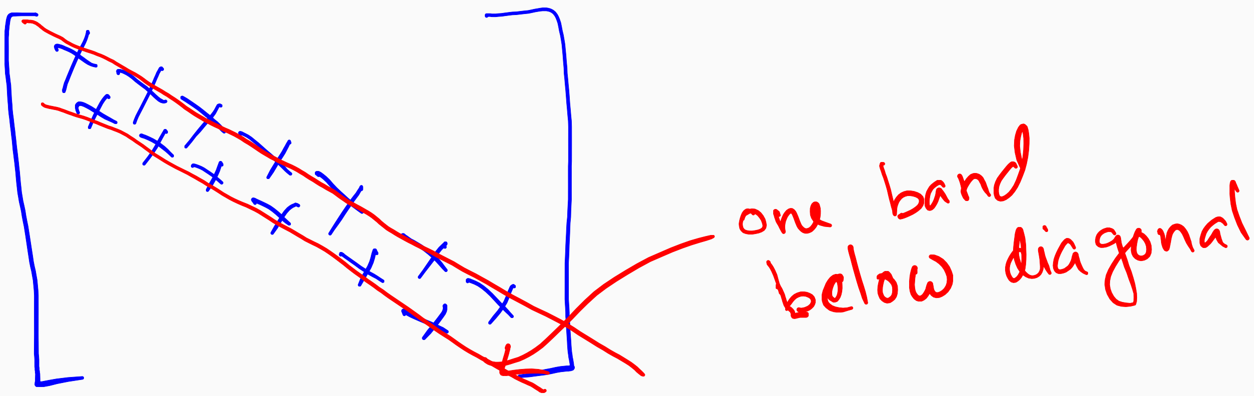
Example:



Lower Bidiagonal:

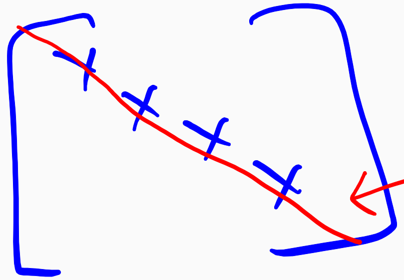
low. bandwidth = 0, upp. bandwidth = 1

Example:



Diagonal Matrix:

low. bandwidth = 0, upp. bandwidth = 0

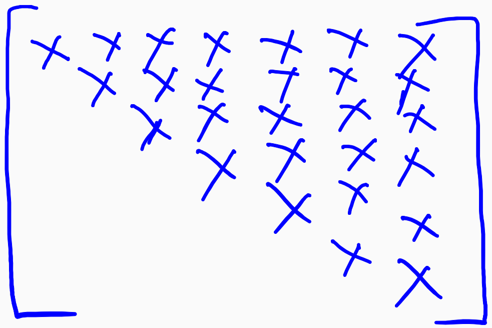


only diagonal band present

Upper Triangular Matrices:

low. bandwidth = 0, upp. bandwidth = $n-1$

Note: $A \in \mathbb{R}^{m \times n}$

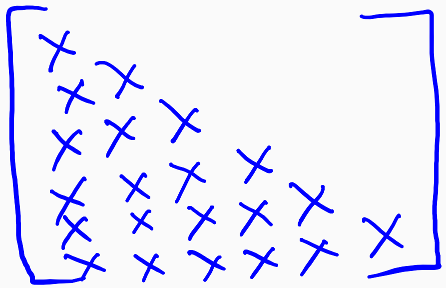


$n-1=6$

7×7

Lower Triangular Matrices:

low. bandwidth = $m-1$, upp. bandwidth = 0

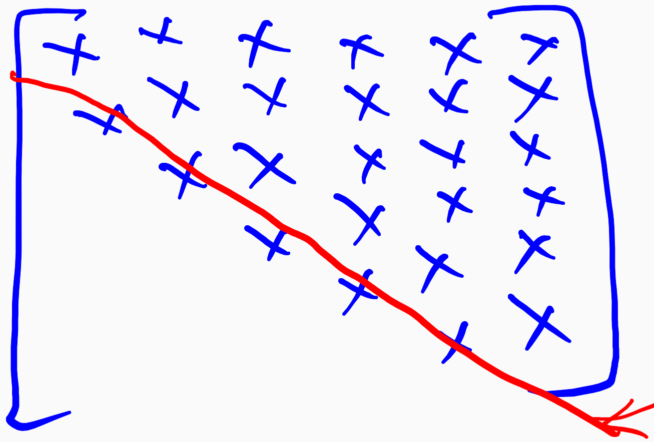


$m-1=6$

6×6

Upper Hessenberg Matrices:

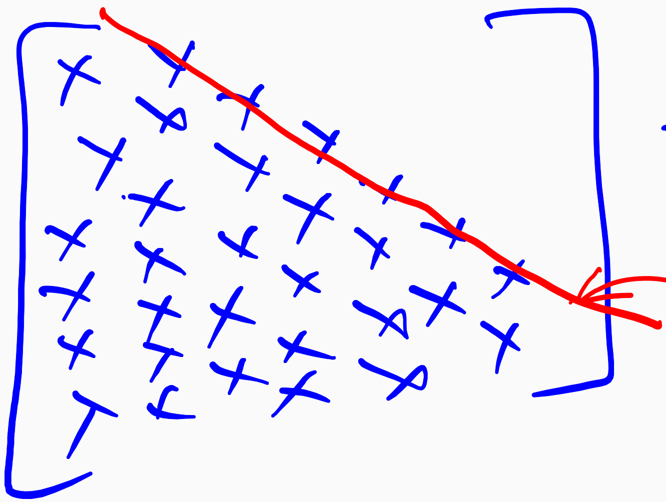
low. bandwidth = 1, upp. bandwidth = $n-1$



Upper Hessenberg is upper triangular and has exactly one lower band

Lower Hessenberg Matrices:

low. bandwidth = $m-1$, upp. bandwidth = 1



Lower Hessenberg is lower triangular and has exactly one upper band

Triangular Matrix Multiplication

Recall row formulation

Example: Let $A, B \in \mathbb{R}^{n\times n}$ be upp. triangular

Algorithm for $C = C + AB$

$$\begin{pmatrix} -a_1^T- \\ -a_2^T- \\ \vdots \\ -a_n^T- \end{pmatrix} \begin{bmatrix} | & | & & | \\ b_1 & b_2 & \cdots & b_n \\ | & | & & | \end{bmatrix}$$

$$\begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ 0 & a_{22} & \cdots & a_{2n} \\ \vdots & \cdots & \ddots & \vdots \\ 0 & \cdots & \cdots & a_{nn} \end{pmatrix} \begin{pmatrix} b_{11} & b_{12} & \cdots & b_{1n} \\ 0 & b_{22} & \cdots & b_{2n} \\ \vdots & \cdots & \ddots & \vdots \\ 0 & \cdots & \cdots & b_{nn} \end{pmatrix}$$

i j k k 0 j

Since, $a_i^T b_j = 0$ for $j < i$

so, $\boxed{j = i:n}$

no need to run k loop for zeros

k starts from i

$$= \begin{pmatrix} \boxed{a_{11}b_{11}} & a_{11}b_{12} + a_{12}b_{22} & \cdots & a_{11}b_{1n} + \cdots + a_{1n}b_{nn} \\ 0 & a_{22}b_{22} & \cdots & a_{22}b_{2n} + a_{23}b_{3n} + \cdots + a_{2n}b_{nn} \\ \vdots & \cdots & \ddots & \vdots \\ 0 & 0 & \cdots & a_{nn}b_{nn} \end{pmatrix}$$

0

k stops at j

So, $k = i:j$

Triangular Matrix Multiplication

Example: Let $A, B \in \mathbb{R}^{n \times n}$ be upp. triangular

Find $C = C + AB$

for $i = 1 : n$ do

  for $j = i : n$ do

    for $k = i : j$ do

      $C(i,j) = C(i,j) + A(i,k)B(k,j)$

    end for

  end for

end for

Cost:

n $n-i$

1 Flop

1 Flop $= 2$ Flops

$i = 1$

$j = 1 : n$

$k = 1 : 1$

$k = 1 : 2$

$\vdots$

$2(j-i)$

$j = i+2$

$$\sum_{i=1}^{n} \sum_{j=i}^{n} 2(j-i) = \sum_{i=1}^{n} \sum_{j=1}^{n-i+1} 2(j-1) \approx O(n^2)$$

$2(i+1-i)$

Band Storage

Given a banded matrix A with lower bandwidth p, and upper bandwidth q, such matrix can be stored as $(p+q+1) \times n$ matrix. Example:

q = 2

i-j=1

p=1

i-j=-2

i-j=-1

i-j=0

↗ use these invariants

$$A = \begin{bmatrix} a_{11} & a_{12} & a_{13} & 0 & 0 & 0 \\ a_{21} & a_{22} & a_{23} & a_{24} & 0 & 0 \\ 0 & a_{32} & a_{33} & a_{34} & a_{35} & 0 \\ 0 & 0 & a_{43} & a_{44} & a_{45} & a_{46} \\ 0 & 0 & 0 & a_{54} & a_{55} & a_{56} \\ 0 & 0 & 0 & 0 & a_{65} & a_{66} \end{bmatrix}$$

$i \to i-j+q+1$

$j \to j$ (col indices don't change)

$(i, j) \to (i-j+q+1, j)$

$$A.band = \begin{bmatrix} * & * & a_{13} & a_{24} & a_{35} & a_{46} \\ * & a_{12} & a_{23} & a_{34} & a_{45} & a_{56} \\ a_{11} & a_{22} & a_{33} & a_{44} & a_{55} & a_{66} \\ a_{21} & a_{32} & a_{43} & a_{54} & a_{65} & * \end{bmatrix}$$

Relate a_{ij} of A.band to a_{ij} of A

$$A = \begin{bmatrix} a_{11} & a_{12} & a_{13} & 0 & 0 & 0 \\ a_{21} & a_{22} & a_{23} & a_{24} & 0 & 0 \\ 0 & a_{32} & a_{33} & a_{34} & a_{35} & 0 \\ 0 & 0 & a_{43} & a_{44} & a_{45} & a_{46} \\ 0 & 0 & 0 & a_{54} & a_{55} & a_{56} \\ 0 & 0 & 0 & 0 & a_{65} & a_{65} \end{bmatrix}$$

$$A.band = \begin{bmatrix} * & * & a_{13} & a_{24} & a_{35} & a_{46} \\ * & a_{12} & a_{23} & a_{34} & a_{45} & a_{56} \\ a_{11} & a_{22} & a_{33} & a_{44} & a_{55} & a_{66} \\ a_{21} & a_{32} & a_{43} & a_{54} & a_{65} & * \end{bmatrix}$$

$$a_{ij} = A.band(i - j + 1 + \overset{q}{\cancel{p}}, j),$$

typo!

where a_{ij} is the $(i, j)th$ entry of A

Banded Gaxpy

Let $A \in \mathbb{R}^{n \times n}$ has lower bandwidth p, and upper bandwidth q, and it is stored in the $A.band$ format. Let $x, y \in \mathbb{R}^n$, then

Exercise: Implement this $y \leftarrow y + Ax$

for $j = 1 : n$ **do**

$\alpha_1 = \max(1, j - q), \quad \alpha_2 = \min(n, j + p)$

$\beta_1 = \max(1, q + 2 - j), \quad \beta_2 = \beta_1 + \alpha_2 - \alpha_1$

$y(\alpha_1 : \alpha_2) = y(\alpha_1 : \alpha_2) + A.band(\beta_1 : \beta_2, j)x(j)$

end for

Symmetry in matrices

Symmetric Matrix:

$A = A^T$

Hermitian Matrix:

Later

Skew-Symmetric Matrix:

$A = -A^T$

Skew-Hermitian Matrix:

Later

Storage: $n^2/2$

Storing Symmetric Matrices

Symmetric matrices can be stored compactly as a vector

$$A = \begin{bmatrix} 1 & 2 & 3 \\ 2 & 4 & 5 \\ 3 & 5 & 6 \end{bmatrix}$$

can be stored as

$$A.vec = [1, 2, 3, 4, 5, 6]$$

We have

$$A.vec(\underbrace{(n - j/2)(j - 1) + i}_{\text{How?}}) = a_{ij}, \quad 1 \leq i, j \leq n$$

Permutation Matrices and Identity Matrices

- I_n : $n \times n$ identity matrix
- e_i : *ith* column of I_n
- **Permutation Matrix:** Rows or cols of I_n are reordered

Example: $P = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix}$

P can be stored as vector: $[2, 1, 4, 3]$

$Px = P \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} = \begin{bmatrix} x_2 \\ x_1 \\ x_4 \\ x_3 \end{bmatrix}$

Storage of Permutation Matrix: $O(n)$

Block Matrices and Algorithms

Block Diagonal

$$A = \begin{bmatrix} A_{11} & 0 & \cdots & 0 \\ 0 & A_{22} & \cdots & \vdots \\ \vdots & \cdots & \ddots & \vdots \\ \vdots & \cdots & \cdots & A_{nn} \end{bmatrix}$$

Block Lower triangular

$$A = \begin{bmatrix} A_{11} & 0 & \cdots & 0 \\ A_{21} & A_{22} & \cdots & \vdots \\ \vdots & \cdots & \ddots & \vdots \\ A_{n1} & A_{n2} & \cdots & A_{nn} \end{bmatrix}$$

Remark: Similarly for block upper triangular

Block Matrix Operations

Transpose of a block matrix:

$$\begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix}^T = \begin{bmatrix} A_{11}^T & A_{21}^T \\ A_{12}^T & A_{22}^T \end{bmatrix}$$

Addition of two block matrices:

$$\begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix} + \begin{bmatrix} B_{11} & B_{12} \\ B_{21} & B_{22} \end{bmatrix} = \begin{bmatrix} A_{11}+B_{11} & A_{12}+B_{12} \\ A_{21}+B_{21} & A_{22}+B_{22} \end{bmatrix}$$

Multiplication of two block matrices

$$
\begin{bmatrix}
A_{11} & A_{12} & A_{13} & \cdots & A_{1n} \\
A_{21} & A_{22} & A_{23} & \cdots & A_{2n} \\
\vdots & \cdots & \ddots & \cdots & \vdots \\
A_{n1} & \cdots & \cdots & \cdots & A_{nn}
\end{bmatrix}
\begin{bmatrix}
B_{11} & B_{12} & B_{13} & \cdots & B_{1n} \\
B_{21} & B_{22} & B_{23} & \cdots & B_{2n} \\
\vdots & \cdots & \ddots & \cdots & \vdots \\
B_{n1} & \cdots & \cdots & \cdots & B_{nn}
\end{bmatrix}
$$

$$
= \begin{bmatrix}
C_{11} & C_{12} & C_{13} & \cdots & C_{1n} \\
C_{21} & C_{22} & C_{23} & \cdots & C_{2n} \\
\vdots & \cdots & \ddots & \cdots & \vdots \\
C_{n1} & \cdots & \cdots & \cdots & C_{nn}
\end{bmatrix},
$$

where

$$
C_{ij} = \sum_{k=1}^{n} A_{ik} B_{kj}
$$

Kronecker or Tensor Product

If B and C are two matrices, then $B \otimes C$ is a block matrix with blocks $b_{ij}C$

Examples:

$$B = \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix}_{2\times 2}, \quad C = \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{bmatrix}_{2\times 3}$$

$$B \otimes C = \begin{bmatrix} 1\begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{bmatrix} & 2\begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{bmatrix} \\ 3\begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{bmatrix} & 4\begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{bmatrix} \end{bmatrix}_{4\times 6}$$

Properties of Kronecker or Tensor Product

$$B = \begin{bmatrix} 1 & 0 \\ 0 & 2 \end{bmatrix}, \quad C = \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{bmatrix}$$

$$B \otimes C = \begin{bmatrix} \underbrace{1\begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{bmatrix}}_{A_{11}} & 0 \\ 0 & \underbrace{2\begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{bmatrix}}_{A_{22}} \end{bmatrix}$$

blk diag.

1. B is diagonal, then $B \otimes C$ is block diagonal
2. B is tridiagonal, then $B \otimes C$ is block tridiagonal
3. B is lower tridiagonal, then $B \otimes C$ is block lower triangular
4. B is upper triangular, then $B \otimes C$ is block upper triangular

Solving $Ax = b$

Solving $Ax = b$ is central to scientific computing. It is also needed in:

- Kernel Ridge Regression
- Second order optimization methods

Steps to solve $Ax = b$:

- Factor the given matrix as follows:

$$A = LU,$$

  where L is lower and U is upper triangular matrices.
- Solve $Ax = b$ by solving $LUx = b$ in following steps:
  1. Solve $Ly = b$ called forward sweep
  2. Solve $Ux = y$ called backward sweep

Forward and Backward Sweeps

Forward Sweep:

$$\begin{bmatrix} \ell_{11} & 0 & \cdots & 0 \\ \ell_{21} & \ell_{22} & \cdots & \vdots \\ \vdots & \cdots & \ddots & 0 \\ \ell_{n1} & \cdots & \cdots & \ell_{nn} \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_n \end{bmatrix}$$

0

$l_{11} y_1 = b_1 \rightarrow$ get y_1

$l_{21} y_1 + l_{22} y_2 = b_2 \rightarrow$ get y_2 (y_1 is known)

$\vdots$

$l_{n1} y_1 + l_{n2} y_2 + \cdots + l_{nn} y_n = b_n \rightarrow$ get y_n

Forward and Backward Sweeps

Backward Sweep:

$$\begin{bmatrix} u_{11} & u_{12} & \cdots & u_{1n} \\ 0 & u_{22} & \cdots & \vdots \\ \vdots & \cdots & \cdots & \vdots \\ 0 & \cdots & 0 & u_{nn} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix}$$

$u_{n-1,n-1}$ $u_{n-1,n}$ 0

$u_{nn} x_n = y_n \Rightarrow$ get x_n

$u_{n-1,n-1} x_{n-1} + u_{n-1,n} x_n = y_{n-1}$ get x_{n-1}

= = =

$u_{11} x_1 + u_{12} x_2 + \cdots + u_{1n} x_n = y_1 \Rightarrow$ get x_1

Algorithms for Forward and Backward Sweeps

Algorithm for forward sweep:

$y_1 = b_1 / l_{11}$
for $i = 1 : n$
  $t = b_i$
  for $j = 1 : i-1$
    $t = t - l_{ij} y_j$
  end
  $y_i = t / l_{ii}$
end

Algorithm for backward sweep:

Try!

Algebra of Triangular Matrices

1. Inverse of an upper (lower) triangular matrix is upper (lower) triangular
2. Product of two upper (lower) triangular matrices is upper (lower) triangular
3. Inverse of an unit upper (lower) triangular matrix is a unit upper (lower) triangular
4. Product of two unit upper (lower) triangular matrices is a unit upper (lower) triangular

Solving simultaneous linear systems: Algebraic way

Find x_1 and x_2 such that

Eliminate x_1 →
$$\begin{cases} 3x_1 + 5x_2 = 9 & \to (1) \times 2 \\ 6x_1 + 7x_2 = 4 & \to (2) \end{cases}$$

(−) (−)

$$\cancel{0x_1} + 3x_2 = 14$$

$$x_2 = 14/3$$

$$A \to \begin{bmatrix} 3 & 5 \\ 6 & 7 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 9 \\ 4 \end{bmatrix}$$

$$U \to \begin{bmatrix} 6 & 10 \\ 0 & 3 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 8 \\ 14 \end{bmatrix}$$

Note: Here A is reduced to U, a triangular matrix, but $A \neq U$

Quiz what is L s.t $A = L \cdot U$?