Sparse Matrices

Ned Nedialkov

McMaster University

14 March 2023

Outline

- Coordinate format (COO)
- Compressed Sparse Row (CSR)
- Compressed Sparse Column (CSC)
- Matrix times vector
  - COO $y = Ax$
  - CSR, CSC $y = Ax$
- $y = A^T x$
  - CSR $y = A^T x$

Example 1.

$$A = \begin{bmatrix} 0 & 1 & 0 & 2 & 0 \\ 3 & 0 & 0 & 0 & 0 \\ 0 & 0 & 4 & 0 & 0 \\ 5 & 0 & 6 & 7 & 0 \\ 0 & 0 & 0 & 0 & 8 \end{bmatrix}$$

Store nonzeros only.

Representations

- Coordinate or triplet format, COO
- Compressed Sparse Row (CSR), also called Compressed Row Storage (CRS)
- Compressed Sparse Column (CSC), also called Compressed Column Storage (CCS)

Coordinate format (COO)

Store nonzero a_{ij} as the triplet (i, j, a_{ij}).

Example 2.

$$A = \begin{array}{c} \\ 1 \\ 2 \\ 3 \\ 4 \\ 5 \end{array} \begin{array}{c} \begin{array}{ccccc} 1 & 2 & 3 & 4 & 5 \end{array} \\ \begin{bmatrix} & 1 & & 2 & \\ 3 & & & & \\ & & 4 & & \\ 5 & & 6 & 7 & \\ & & & & 8 \end{bmatrix} \end{array}$$

row_index	1	1	2	3	4	4	4	5
col_index	2	4	1	3	1	3	4	5
value	1	2	3	4	5	6	7	8

array	stores
row_index	row indexes
col_index	column indices
value	nonzeros

Compressed Sparse Row (CSR)

From COO, compress the row indices as a row_pointer array.

Example 3.

row_index	1	1	2	3	4	4	4	5	
offset	1	2	3	4	5	6	7	8	9
row_pointer	1	3	4	5	8	9			

Assume A is $m \times n$. Denote the number of nonzeros by nnz.

- row_pointer$(i+1)$ − row_pointer(i) is number of nonzeros in row i
- nnz = row_pointer$(m+1)$ − row_pointer(1) is total number of nonzeros.

Example 3. cont.

$$A = \begin{array}{c} \\ 1 \\ 2 \\ 3 \\ 4 \\ 5 \end{array} \begin{array}{c} \begin{array}{ccccc} 1 & 2 & 3 & 4 & 5 \end{array} \\ \begin{bmatrix} & 1 & & 2 & \\ 3 & & & & \\ & & 4 & & \\ 5 & & 6 & 7 & \\ & & & & 8 \end{bmatrix} \end{array}$$

CSR

row_pointer

1	3	4	5	8	9

col_index

2	4	1	3	1	3	4	5

value

1	2	3	4	5	6	7	8

Compressed Sparse Column (CSC)

Compress column indices into a col_pointer array.

Example 4.

$$A = \begin{array}{c} \\ 1 \\ 2 \\ 3 \\ 4 \\ 5 \end{array} \begin{array}{c} \begin{array}{ccccc} 1 & 2 & 3 & 4 & 5 \end{array} \\ \begin{bmatrix} & 1 & & 2 & \\ 3 & & & & \\ & & 4 & & \\ 5 & & 6 & 7 & \\ & & & & 8 \end{bmatrix} \end{array}$$

CSC

col_pointer

1	3	4	6	8	9

offset	1	2	3	4	5	6	7	8	9
row_index	2	4	1	3	4	1	4	5	
value	3	5	1	4	6	2	7	8	

Matrix times vector

Consider $y = Ax$, where A is $m \times n$.

$$y_i = \sum_{j=1}^{n} a_{ij} x_j, \quad i = 1 : m.$$

In the algorithms that follow, assume that initially the entries of y are set to 0.

Algorithm 1 $(y = Ax)$.

for $i = 1 : m$
    for $j = 1 : n$
        $y(i) \mathrel{+}= A(i, j) * x(j)$

COO $y = Ax$

If A is in COO format, $y = Ax$ can be computed as

Algorithm 2 (COO $y = Ax$).

for $k = 1 : \text{nnz}$

$i = \text{row\_index}(k)$

$j = \text{col\_index}(k)$

$y(i) \mathrel{+}= \text{value}(k) * x(j)$

CSR, CSC $y = Ax$

Algorithm 3 (*CSR* $y = Ax$).

for $i = 1 : m$
    for $k =$ row_pointer(i) : row_pointer$(i+1) - 1$
        $j =$ col_index(k)
        $y(i)$ += value$(k) * x(j)$

Algorithm 4 (*CSC* $y = Ax$).

for $j = 1 : n$
    for $k =$ col_pointer(j) : col_pointer$(j+1) - 1$
        $i =$ row_index(k)
        $y(i)$ += value$(k) * x(j)$

$y = A^T x$

Consider $y = A^T x$, where A is $m \times n$; A^T is $n \times m$.

$$y_i = \sum_{j=1}^{m} a_{ji} x_j, \quad i = 1 : n$$

Algorithm 5 ($y = A^T x$).

for $i = 1 : n$
    for $j = 1 : m$
        $y(i) \mathrel{+}= A(j, i) * x(j)$

Inefficient if we process by rows. Swap the loops.

Algorithm 6 ($y = A^T x$).

for $j = 1 : m$
    for $i = 1 : n$
        $y(i) \mathrel{+}= A(j, i) * x(j)$

Algorithm 7 (*CSR* $y = A^T x$).

for $j = 1 : m$
    for $k =$ row_pointer(j) : row_pointer$(j + 1) - 1$
        $i =$ col_index(k)
        $y(i)$ += value$(k) * x(j)$