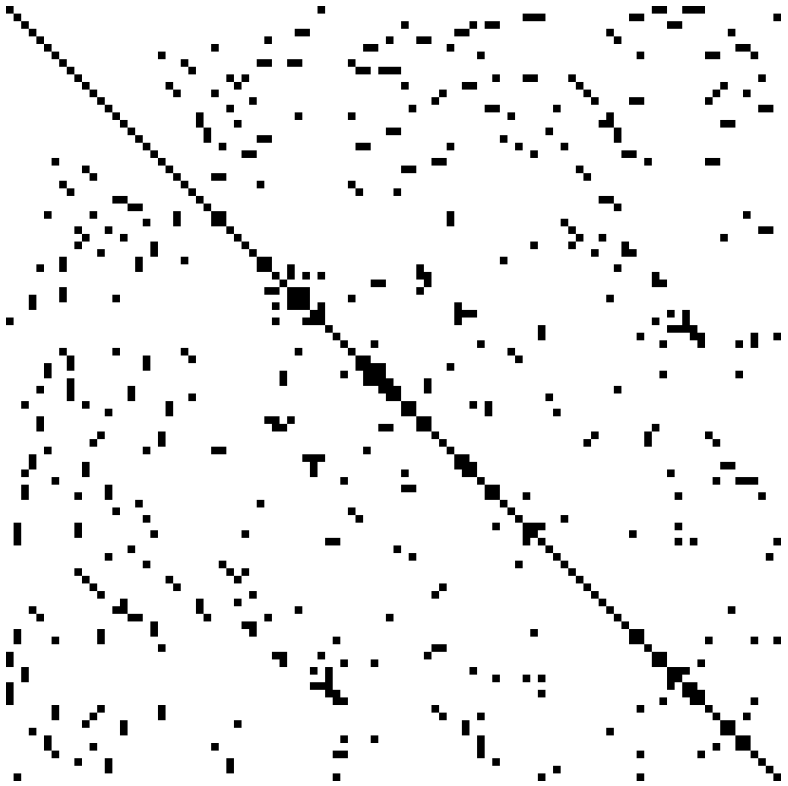
Sparse Matrix Data Structures

Shusen Wang

Stevens Institute of Technology

http://wangshusen.github.io/

Sparse Matrices



Example of sparse matrix

- **Sparse matrix:** A matrix in which most elements are zeros.
- **Question:** How to store a sparse matrix?
- **Bad solution:** As a dense matrix.
- **Good solution:** Storing only the nonzero elements and their indices.

Formats

- Triplet formats (aka Coordinate List) is easy to understand.
- CSR and CSC formats are more efficient.

Triplet Formats (aka Coordinate Lists)

Triplet Format (Row-Major)

$$\mathbf{A} = \begin{bmatrix} 10 & 5.5 & 0 & 0 & 0 & 0 \\ 0 & 0 & 30 & 40 & 0 & 7.1 \\ 0 & 2 & 0 & 0 & 70 & 0 \\ 0 & 0 & 0 & 9.2 & 0 & 26 \end{bmatrix}$$

Triplet Format (Row-Major)

$$\mathbf{A} = \begin{bmatrix} 10 & 5.5 & 0 & 0 & 0 & 0 \\ 0 & 0 & 30 & 40 & 0 & 7.1 \\ 0 & 2 & 0 & 0 & 70 & 0 \\ 0 & 0 & 0 & 9.2 & 0 & 26 \end{bmatrix}$$

Value:

10	5.5	30	40	7.1	2	70	9.2	26

Triplet Format (Row-Major)

$$A = \begin{bmatrix} 10 & 5.5 & 0 & 0 & 0 & 0 \\ 0 & 0 & 30 & 40 & 0 & 7.1 \\ 0 & 2 & 0 & 0 & 70 & 0 \\ 0 & 0 & 0 & 9.2 & 0 & 26 \end{bmatrix}$$

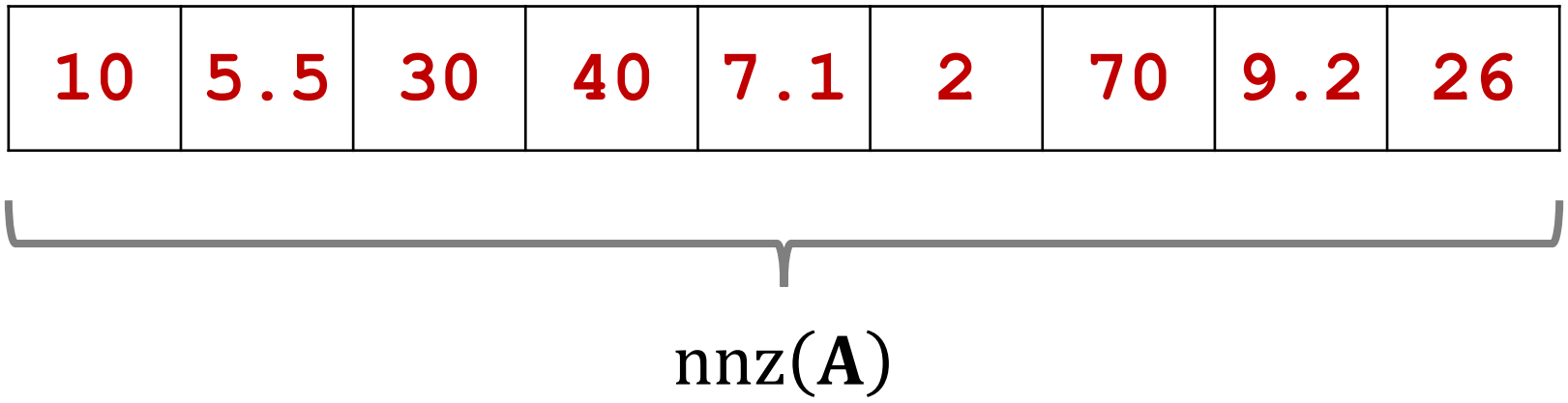
Value:

10	5.5	30	40	7.1	2	70	9.2	26

Triplet Format (Row-Major)

$$\mathbf{A} = \begin{bmatrix} 10 & 5.5 & 0 & 0 & 0 & 0 \\ 0 & 0 & 30 & 40 & 0 & 7.1 \\ 0 & 2 & 0 & 0 & 70 & 0 \\ 0 & 0 & 0 & 9.2 & 0 & 26 \end{bmatrix}$$

Value:



Triplet Format (Row-Major)

$$A = \begin{bmatrix} 10 & 5.5 & 0 & 0 & 0 & 0 \\ 0 & 0 & 30 & 40 & 0 & 7.1 \\ 0 & 2 & 0 & 0 & 70 & 0 \\ 0 & 0 & 0 & 9.2 & 0 & 26 \end{bmatrix}$$

Value:	10	5.5	30	40	7.1	2	70	9.2	26
Row Index:	0	0	1	1	1	2	2	3	3
Col Index:	0	1	2	3	5	1	4	3	5

Triplet Format (Row-Major)

$$A = \begin{bmatrix} 10 & 5.5 & 0 & 0 & 0 & 0 \\ 0 & 0 & 30 & 40 & 0 & 7.1 \\ 0 & 2 & 0 & 0 & 70 & 0 \\ 0 & 0 & 0 & 9.2 & 0 & 26 \end{bmatrix}$$

Value:	10	5.5	30	40	7.1	2	70	9.2	26
Row Index:	0	0	1	1	1	2	2	3	3
Col Index:	0	1	2	3	5	1	4	3	5

How to slice a row?

Sparse Matrix: $A = \begin{bmatrix} 10 & 5.5 & 0 & 0 & 0 & 0 \\ 0 & 0 & 30 & 40 & 0 & 7.1 \\ 0 & 2 & 0 & 0 & 70 & 0 \\ 0 & 0 & 0 & 9.2 & 0 & 26 \end{bmatrix}$

Value:	10	5.5	30	40	7.1	2	70	9.2	26
Row Index:	0	0	1	1	1	2	2	3	3
Col Index:	0	1	2	3	5	1	4	3	5

How to slice a row?

Sparse Matrix: $\mathbf{A} = \begin{bmatrix} 10 & 5.5 & 0 & 0 & 0 & 0 \\ 0 & 0 & 30 & 40 & 0 & 7.1 \\ 0 & 2 & 0 & 0 & 70 & 0 \\ 0 & 0 & 0 & 9.2 & 0 & 26 \end{bmatrix}$

Value:	10	5.5	30	40	7.1	2	70	9.2	26
Row Index:	0	0	1	1	1	2	2	3	3
Col Index:	0	1	2	3	5	1	4	3	5

How to slice a row?

Sparse Matrix: $A = \begin{bmatrix} 10 & 5.5 & 0 & 0 & 0 & 0 \\ 0 & 0 & 30 & 40 & 0 & 7.1 \\ 0 & 2 & 0 & 0 & 70 & 0 \\ 0 & 0 & 0 & 9.2 & 0 & 26 \end{bmatrix}$

Traversing a row is fast.

Value:	10	5.5	30	40	7.1	2	70	9.2	26
Row Index:	0	0	1	1	1	2	2	3	3
Col Index:	0	1	2	3	5	1	4	3	5

How to slice a column?

Sparse Matrix: $\mathbf{A} = \begin{bmatrix} 10 & 5.5 & 0 & 0 & 0 & 0 \\ 0 & 0 & 30 & 40 & 0 & 7.1 \\ 0 & 2 & 0 & 0 & 70 & 0 \\ 0 & 0 & 0 & 9.2 & 0 & 26 \end{bmatrix}$

Value:	10	5.5	30	40	7.1	2	70	9.2	26
Row Index:	0	0	1	1	1	2	2	3	3
Col Index:	0	1	2	3	5	1	4	3	5

How to slice a column?

Sparse Matrix: $\mathbf{A} = \begin{bmatrix} 10 & 5.5 & 0 & 0 & 0 & 0 \\ 0 & 0 & 30 & 40 & 0 & 7.1 \\ 0 & 2 & 0 & 0 & 70 & 0 \\ 0 & 0 & 0 & 9.2 & 0 & 26 \end{bmatrix}$

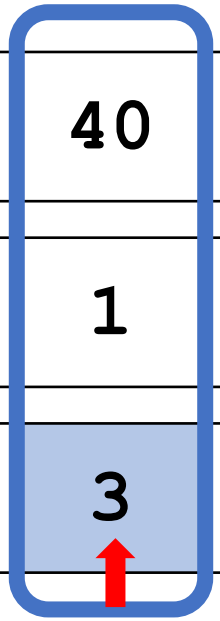
Value:	10	5.5	30	40	7.1	2	70	9.2	26
Row Index:	0	0	1	1	1	2	2	3	3
Col Index:	0	1	2	3	5	1	4	3	5

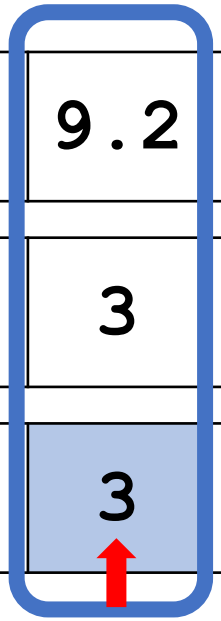
How to slice a column?

Sparse Matrix: $A = \begin{bmatrix} 10 & 5.5 & 0 & 0 & 0 & 0 \\ 0 & 0 & 30 & 40 & 0 & 7.1 \\ 0 & 2 & 0 & 0 & 70 & 0 \\ 0 & 0 & 0 & 9.2 & 0 & 26 \end{bmatrix}$

Traversing a column is slow.

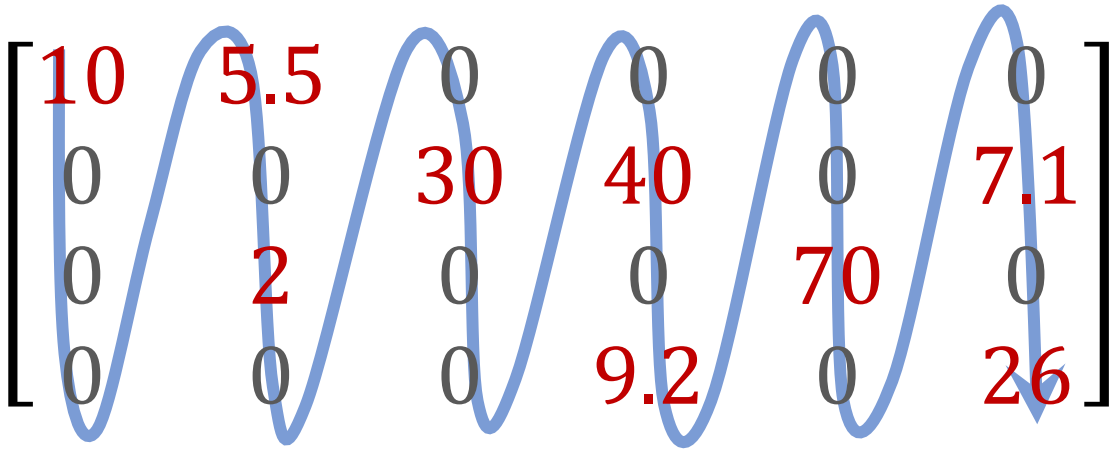
Value:	10	5.5	30	40	7.1	2	70	9.2	26
Row Index:	0	0	1	1	1	2	2	3	3
Col Index:	0	1	2	3	5	1	4	3	5





Triplet Format (Column-Major)

$$\mathbf{A} = \begin{bmatrix} 10 & 5.5 & 0 & 0 & 0 & 0 \\ 0 & 0 & 30 & 40 & 0 & 7.1 \\ 0 & 2 & 0 & 0 & 70 & 0 \\ 0 & 0 & 0 & 9.2 & 0 & 26 \end{bmatrix}$$



Triplet Format (Column-Major)

$$\mathbf{A} = \begin{bmatrix} 10 & 5.5 & 0 & 0 & 0 & 0 \\ 0 & 0 & 30 & 40 & 0 & 7.1 \\ 0 & 2 & 0 & 0 & 70 & 0 \\ 0 & 0 & 0 & 9.2 & 0 & 26 \end{bmatrix}$$

Value:

10	5.5	2	30	40	9.2	70	7.1	26

Triplet Format (Column-Major)

$$\mathbf{A} = \begin{bmatrix} 10 & 5.5 & 0 & 0 & 0 & 0 \\ 0 & 0 & 30 & 40 & 0 & 7.1 \\ 0 & 2 & 0 & 0 & 70 & 0 \\ 0 & 0 & 0 & 9.2 & 0 & 26 \end{bmatrix}$$

Value:

10	5.5	2	30	40	9.2	70	7.1	26

Triplet Format (Column-Major)

$$A = \begin{bmatrix} 10 & 5.5 & 0 & 0 & 0 & 0 \\ 0 & 0 & 30 & 40 & 0 & 7.1 \\ 0 & 2 & 0 & 0 & 70 & 0 \\ 0 & 0 & 0 & 9.2 & 0 & 26 \end{bmatrix}$$

Value:	10	5.5	2	30	40	9.2	70	7.1	26
Row Index:	0	0	2	1	1	3	2	1	3
Col Index:	0	1	1	2	3	3	4	5	5

Triplet Format (Column-Major)

$$\mathbf{A} = \begin{bmatrix} 10 & 5.5 & 0 & 0 & 0 & 0 \\ 0 & 0 & 30 & 40 & 0 & 7.1 \\ 0 & 2 & 0 & 0 & 70 & 0 \\ 0 & 0 & 0 & 9.2 & 0 & 26 \end{bmatrix}$$

Value:	10	5.5	2	30	40	9.2	70	7.1	26
Row Index:	0	0	2	1	1	3	2	1	3
Col Index:	0	1	1	2	3	3	4	5	5

Memory Cost

- 8 Bytes for a double-precision floating-point number (a value).
- 4 Bytes for a long integer (an index).

Memory Cost

- 8 Bytes for a double-precision floating-point number (a value).
- 4 Bytes for a long integer (an index).
- Memory cost (Bytes) of triplet format:

$$(8 + 4 + 4) \cdot \mathrm{nnz}(\mathbf{A}) = 16 \cdot \mathrm{nnz}(\mathbf{A}).$$

- Memory cost (Bytes) of an $m \times n$ dense matrix:

$$8mn.$$

- If over 50% elements are zeros, then triplet format saves memory.

Compressed Sparse Row (CSR)

Compressed Sparse Row (CSR)

Sparse Matrix: $A = \begin{bmatrix} 10 & 5.5 & 0 & 0 & 0 & 0 \\ 0 & 0 & 30 & 40 & 0 & 7.1 \\ 0 & 2 & 0 & 0 & 70 & 0 \\ 0 & 0 & 0 & 9.2 & 0 & 26 \end{bmatrix}$

Row Index:

0	2	5	7	9

Col Index:

0	1	2	3	5	1	4	3	5

Value:

10	5.5	30	40	7.1	2	70	9.2	26

Compressed Sparse Row (CSR)

Sparse Matrix: $A = \begin{bmatrix} 10 & 5.5 & 0 & 0 & 0 & 0 \\ 0 & 0 & 30 & 40 & 0 & 7.1 \\ 0 & 2 & 0 & 0 & 70 & 0 \\ 0 & 0 & 0 & 9.2 & 0 & 26 \end{bmatrix}$

Row Index:	0	2	5	7	9

Col Index:	0	1	2	3	5	1	4	3	5

Value:	10	5.5	30	40	7.1	2	70	9.2	26

Compressed Sparse Row (CSR)

Sparse Matrix: $A = \begin{bmatrix} 10 & 5.5 & 0 & 0 & 0 & 0 \\ 0 & 0 & 30 & 40 & 0 & 7.1 \\ 0 & 2 & 0 & 0 & 70 & 0 \\ 0 & 0 & 0 & 9.2 & 0 & 26 \end{bmatrix}$

Row Index:

0	2	5	7	9

Col Index:

0	1	2	3	5	1	4	3	5

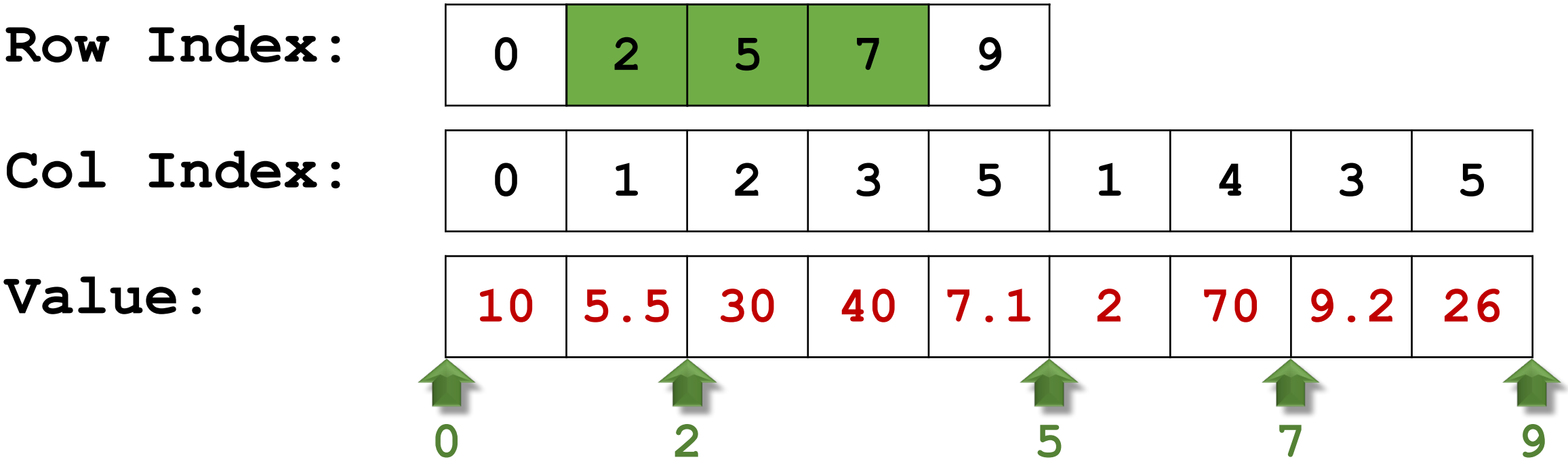
Value:

10	5.5	30	40	7.1	2	70	9.2	26

0 9

Compressed Sparse Row (CSR)

Sparse Matrix: $A = \begin{bmatrix} 10 & 5.5 & 0 & 0 & 0 & 0 \\ 0 & 0 & 30 & 40 & 0 & 7.1 \\ 0 & 2 & 0 & 0 & 70 & 0 \\ 0 & 0 & 0 & 9.2 & 0 & 26 \end{bmatrix}$



Compressed Sparse Row (CSR)

Sparse Matrix: $A = \begin{bmatrix} 10 & 5.5 & 0 & 0 & 0 & 0 \\ 0 & 0 & 30 & 40 & 0 & 7.1 \\ 0 & 2 & 0 & 0 & 70 & 0 \\ 0 & 0 & 0 & 9.2 & 0 & 26 \end{bmatrix}$

Row Index:

0	2	5	7	9

Col Index:

0	1	2	3	5	1	4	3	5

Value:

10	5.5	30	40	7.1	2	70	9.2	26

Compressed Sparse Row (CSR)

Sparse Matrix: $A =$

$$\begin{bmatrix} 10 & 5.5 & 0 & 0 & 0 & 0 \\ 0 & 0 & 30 & 40 & 0 & 7.1 \\ 0 & 2 & 0 & 0 & 70 & 0 \\ 0 & 0 & 0 & 9.2 & 0 & 26 \end{bmatrix}$$

Row Index:

0	2	5	7	9

Col Index:

0	1	2	3	5	1	4	3	5

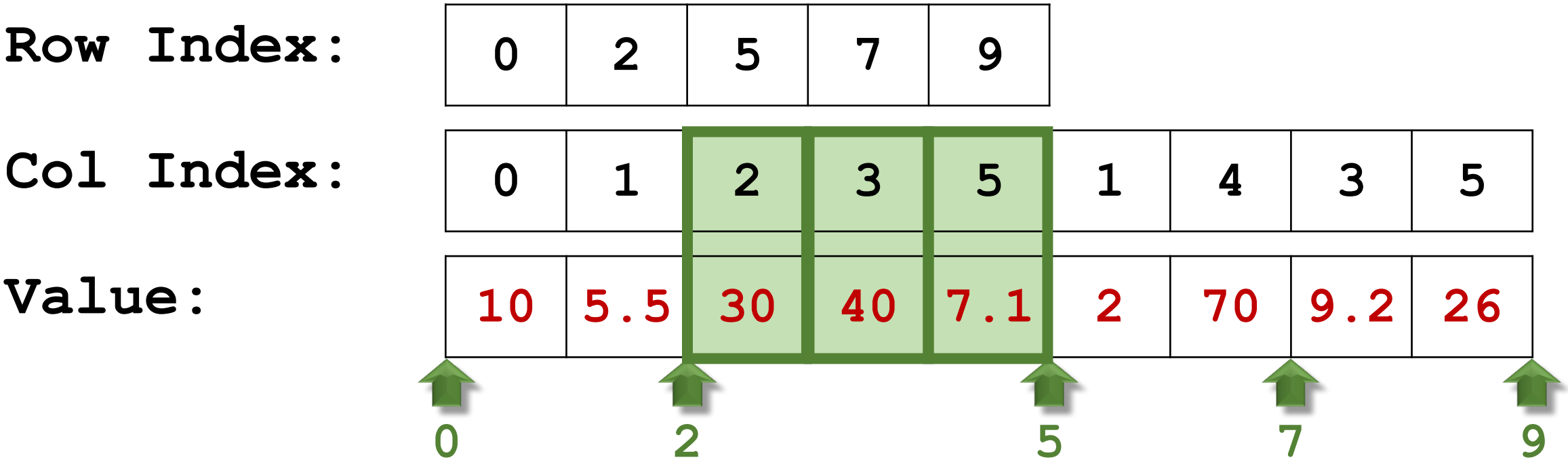
Value:

10	5.5	30	40	7.1	2	70	9.2	26

0 2 5 7 9

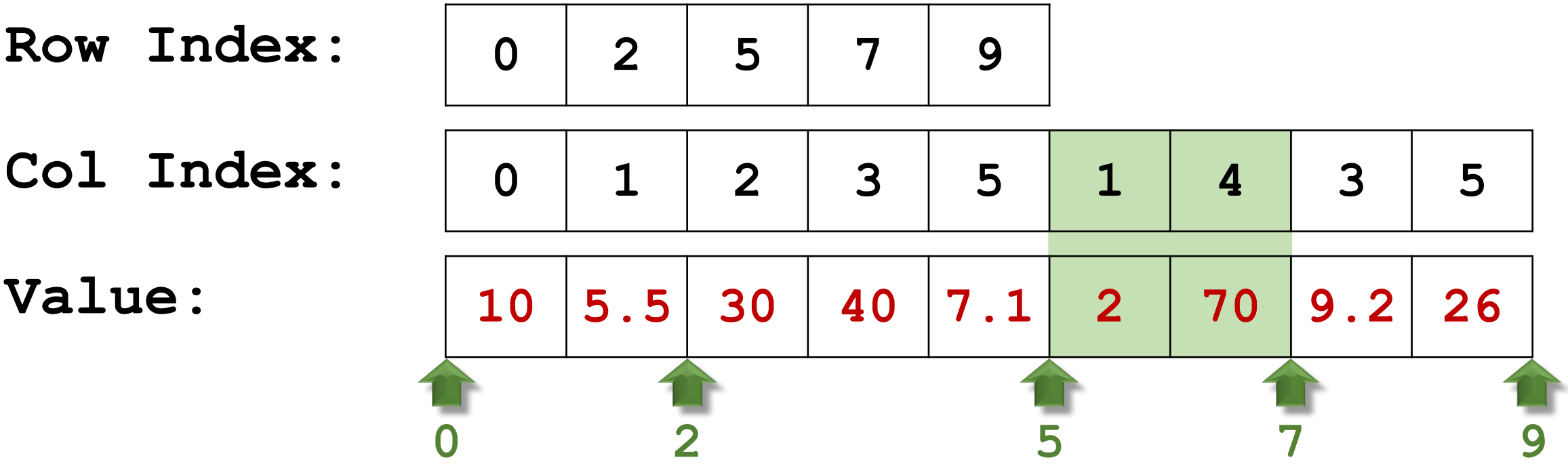
Compressed Sparse Row (CSR)

Sparse Matrix: $A = \begin{bmatrix} 10 & 5.5 & 0 & 0 & 0 & 0 \\ 0 & 0 & 30 & 40 & 0 & 7.1 \\ 0 & 2 & 0 & 0 & 70 & 0 \\ 0 & 0 & 0 & 9.2 & 0 & 26 \end{bmatrix}$



Compressed Sparse Row (CSR)

Sparse Matrix: $A = \begin{bmatrix} 10 & 5.5 & 0 & 0 & 0 & 0 \\ 0 & 0 & 30 & 40 & 0 & 7.1 \\ 0 & 2 & 0 & 0 & 70 & 0 \\ 0 & 0 & 0 & 9.2 & 0 & 26 \end{bmatrix}$



Compressed Sparse Row (CSR)

Sparse Matrix: $A = \begin{bmatrix} 10 & 5.5 & 0 & 0 & 0 & 0 \\ 0 & 0 & 30 & 40 & 0 & 7.1 \\ 0 & 2 & 0 & 0 & 70 & 0 \\ 0 & 0 & 0 & 9.2 & 0 & 26 \end{bmatrix}$

Row Index:

0	2	5	7	9

Col Index:

0	1	2	3	5	1	4	3	5

Value:

10	5.5	30	40	7.1	2	70	9.2	26

0 2 5 7 9

CSR and CSC

- Compressed Sparse Row (**CSR**) has row-major order.
  - Store nonzero elements from the 0^{th} row to the last row.
  - Slicing a row is efficient.
- Compressed Sparse Column (**CSC**) has column-major order.
  - Store nonzero elements from the 0^{th} column to the last column.
  - Slicing a column is efficient.

Triplet Format (Row-Major)

Sparse Matrix: $A = \begin{bmatrix} 10 & 5.5 & 0 & 0 & 0 & 0 \\ 0 & 0 & 30 & 40 & 0 & 7.1 \\ 0 & 2 & 0 & 0 & 70 & 0 \\ 0 & 0 & 0 & 9.2 & 0 & 26 \end{bmatrix}$

Row Index:	0	0	1	1	1	2	2	3	3
Col Index:	0	1	2	3	5	1	4	3	5
Value:	10	5.5	30	40	7.1	2	70	9.2	26

Compressed Sparse Row (CSR)

Sparse Matrix: $A = \begin{bmatrix} 10 & 5.5 & 0 & 0 & 0 & 0 \\ 0 & 0 & 30 & 40 & 0 & 7.1 \\ 0 & 2 & 0 & 0 & 70 & 0 \\ 0 & 0 & 0 & 9.2 & 0 & 26 \end{bmatrix}$

Row Index:

0	2	5	7	9

Col Index:

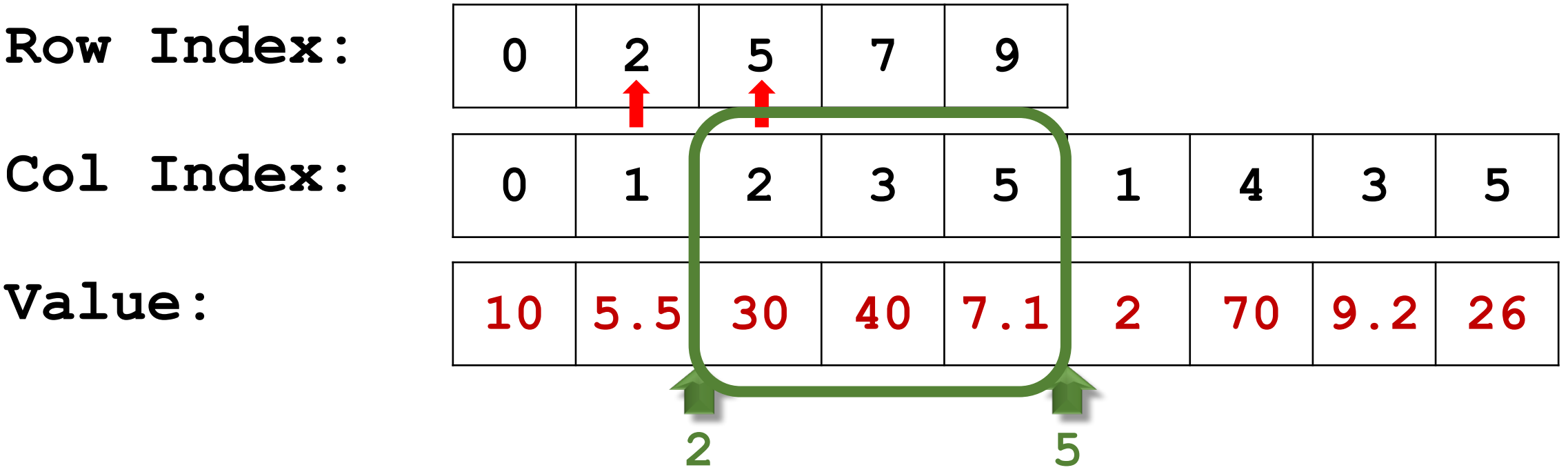
0	1	2	3	5	1	4	3	5

Value:

10	5.5	30	40	7.1	2	70	9.2	26

Compressed Sparse Row (CSR)

Sparse Matrix: $A = \begin{bmatrix} 10 & 5.5 & 0 & 0 & 0 & 0 \\ 0 & 0 & 30 & 40 & 0 & 7.1 \\ 0 & 2 & 0 & 0 & 70 & 0 \\ 0 & 0 & 0 & 9.2 & 0 & 26 \end{bmatrix}$



Triplet to CSR

Row Index:	0	0	1	1	1	2	2	3	3
Col Index:	0	1	2	3	5	1	4	3	5
Value:	10	5.5	30	40	7.1	2	70	9.2	26

Triplet Format (Row-Major)

CSR Format

Row Index:	0

Row Index:	0	0	1	1	1	2	2	3	3
Col Index:	0	1	2	3	5	1	4	3	5
Value:	10	5.5	30	40	7.1	2	70	9.2	26

Triplet Format (Row-Major)

CSR Format

Row Index:	0	2

2

Row Index:	0	0	1	1	1	2	2	3	3
Col Index:	0	1	2	3	5	1	4	3	5
Value:	10	5.5	30	40	7.1	2	70	9.2	26

Triplet Format (Row-Major)

CSR Format

Row Index:	0	2	5

5

Row Index:	0	0	1	1	1	2	2	3	3
Col Index:	0	1	2	3	5	1	4	3	5
Value:	10	5.5	30	40	7.1	2	70	9.2	26

Triplet Format (Row-Major)

CSR Format

Row Index:	0	2	5	7

Row Index:	0	0	1	1	1	2	2	3	3
Col Index:	0	1	2	3	5	1	4	3	5
Value:	10	5.5	30	40	7.1	2	70	9.2	26

Triplet Format (Row-Major)

CSR Format

Row Index:	0	2	5	7	9

9

Row Index:	0	0	1	1	1	2	2	3	3
Col Index:	0	1	2	3	5	1	4	3	5
Value:	10	5.5	30	40	7.1	2	70	9.2	26

Triplet Format (Row-Major)

CSR Format

Row Index:	0	2	5	7	9

Row Index:	0	0	1	1	1	2	2	3	3
Col Index:	0	1	2	3	5	1	4	3	5
Value:	10	5.5	30	40	7.1	2	70	9.2	26

Triplet Format (Row-Major)

CSR Format

Row Index:	0	2	5	7	9

Row Index:	0	0	1	1	1	2	2	3	3
Col Index:	0	1	2	3	5	1	4	3	5
Value:	10	5.5	30	40	7.1	2	70	9.2	26

Triplet Format (Row-Major)

Questions

From Triplet Format to Dense matrix

Value:	9	8.2	29	2	3.1	5	2	1.5	7	10
Row Index:	0	0	0	0	1	1	2	3	3	3
Col Index:	1	3	4	5	0	1	1	2	3	5

Reconstruct the dense matrix:

$$\mathbf{A} = \begin{bmatrix} ? & ? & ? & ? & ? & ? \\ ? & ? & ? & ? & ? & ? \\ ? & ? & ? & ? & ? & ? \\ ? & ? & ? & ? & ? & ? \end{bmatrix}$$

From CSR Format to Dense matrix

Value:	9	8.2	29	2	3.1	5	2	1.5	7	10

Row Index:	0	4	6	7	10

Col Index:	0	2	4	5	0	1	1	2	3	4

Reconstruct the dense matrix:

$$\mathbf{A} = \begin{bmatrix} ? & ? & ? & ? & ? & ? \\ ? & ? & ? & ? & ? & ? \\ ? & ? & ? & ? & ? & ? \\ ? & ? & ? & ? & ? & ? \end{bmatrix}$$

Matrix L1 Norm

Value:	3	2	1	7	4	3	5	1	2
Row Index:	1	1	2	2	2	3	3	4	4
Col Index:	0	1	2	3	5	1	4	3	6

- The 4×6 matrix $\mathbf{A}$ is stored in the triplet format.
- **Question:** What is the ℓ_1-norm of $\mathbf{A}$?
- **Hint:** The matrix ℓ_1-norm is $\left|\left|\mathbf{A}\right|\right|_1 = \sum_{i=1}^{4}\sum_{j=1}^{6}\left|a_{ij}\right|$.

Thank You!

http://wangshusen.github.io/