#### **Esercitazione 3**

# <u>Algebra Lineare</u>

File dell'esercitazione reperibile su:

http://campus.unibo.it/

File dell'esercitazione reperibile su: http://tinyurl.com/CalcoloEs3slide

# Norma 1 e Norma Infinito:

#### **Vettore**

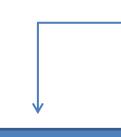
$$x \rightarrow (1 \times n)$$

Norma 1 di un vettore:

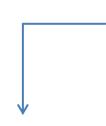
$$||x||_1 = \sum_{i=1}^n |x_i|$$

 Norma Infinito di un vettore:

$$||x||_{\infty} = \max_{i} |x_{i}|$$



Norma 1 vettoriale per colonne



Norma 1 vettoriale per righe

### Matrice

$$A \rightarrow (m \times n)$$

Norma 1 di una matrice:

$$||A||_1 = \max_j \sum_{i=1}^m |a_{ij}|$$

 Norma Infinito di una matrice:

$$||A||_1 = \max_i \sum_{j=1}^n |a_{ij}|$$

# **Comandi Matlab Utili:**

#### m-function in Matlab:

# help

#### Prima riga della funzione:

function [output1, output2,...] = nome\_funzione(input1,input2...)
 es: function norma = calcolo\_norma1(X)

#### Altre funzioni utili:

- function abs(x) -> calcola il valore assoluto di tutti gli elementi di x
- function sum(x) -> somma gli elementi di x

N.B. se x è una matrice, la somma avviene di default per colonne, ma si può fare anche per righe

Usando queste funzioni è possibile realizzare tutte le funzioni richieste con una sola riga di codice per ognuna!!!

# function sum(A):

```
A =

1 3 -5 0
8 -2 4 5
2 0 1 -3
```

• Somma per colonne: dimensione 1

>> sum(A)

ans =

11 1 0 2

>> sum(A,1)

ans =

11 1 0 2

 Somma per righe: dimensione 2

```
>> sum(A,2)

ans =

-1
15
0
```

# Verifica dei Risultati:

# help

# Built-in function per calcolare le norme in Matlab:

• function norm(X)

```
>> help norm
norm    Matrix or vector norm.
    norm(X,2) returns the 2-norm of X.

norm(X) is the same as norm(X,2).

norm(X,1) returns the 1-norm of X.

norm(X,Inf) returns the infinity norm of X.

norm(X,'fro') returns the Frobenius norm of X.
```

# Matrici Sparse in formato CRS

#### **Compressed Sparse Row**

$$A = \begin{bmatrix} 8 & 0 & 0 & 0 & 0 & 0 \\ 0 & 5 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 3 & 5 & 0 \\ 9 & 0 & 1 & 0 & 0 & 0 \\ 0 & 3 & 0 & 0 & 0 & 2 \\ 0 & 0 & 0 & 0 & 7 & 0 \\ 0 & 0 & 9 & 0 & 0 & 3 \end{bmatrix}$$

Matrici Sparse:
Molti elementi nulli



(es. CRS)

# Formato CRS: Compressed Sparse Row

Al posto della matrice A considero 3 vettori:

•  $\mathbf{R} \to (1 \times nnz)$ Contiene tutti gli elementi non nulli di A, memorizzati per righe

•  $\mathbf{c} \to (1 \times nnz)$ Contiene gli indici di colonna degli elementi di R

• I  $\rightarrow$  1×(N+1) Contiene la posizione in R del primo elemento non nullo per ogni riga (N righe totali). L'ultimo elemento contiene il numero degli elementi non nulli (nnz)+1

# Formato CRS: Compressed Sparse Row

$$A = \begin{bmatrix} 8 & 0 & 0 & 0 & 0 & 0 \\ 0 & 5 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 3 & 5 & 0 \\ 9 & 0 & 1 & 0 & 0 & 0 \\ 0 & 3 & 0 & 0 & 0 & 2 \\ 0 & 0 & 0 & 0 & 0 & 7 & 0 \\ 0 & 0 & 9 & 0 & 0 & 3 \end{bmatrix}$$

$$R = \begin{bmatrix} 8 & 5 & 3 & 5 & 9 & 1 & 3 & 2 & 7 & 9 & 3 \end{bmatrix}$$

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

$$I = \begin{bmatrix} 1 & 2 & 3 & 5 & 7 & 9 & 10 & 12 \end{bmatrix} \longrightarrow$$

Totale elementi non nulli + 1

# **Prodotto Matrice CRS x Vettore X**

#### m-function in Matlab:

function prodotto = calcolo\_prodottoCRSvettore(R,C,I,X)

# Formato CRS: Compressed Sparse Row

$$A = \begin{bmatrix} 8 & 0 & 0 & 0 & 0 & 0 \\ 0 & 5 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 3 & 5 & 0 \\ 9 & 0 & 1 & 0 & 0 & 0 \\ 0 & 3 & 0 & 0 & 0 & 2 \\ 0 & 0 & 0 & 0 & 7 & 0 \\ 0 & 0 & 9 & 0 & 0 & 3 \end{bmatrix}$$

$$X = \begin{bmatrix} 1 \\ 0 \\ 2 \\ 3 \\ 1 \\ 1 \end{bmatrix}$$

$$R = \begin{bmatrix} 8 & 5 & 3 & 5 & 9 & 1 & 3 & 2 & 7 & 9 & 3 \end{bmatrix}$$

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

$$I = \begin{bmatrix} 1 & 2 & 3 & 5 & 7 & 9 & 10 & 12 \end{bmatrix}$$

R.\*X(C)

# Prodotto Matrice CRS e Vettore X

#### m-function in Matlab:

function prodotto = calcolo\_prodottoCRSvettore(R,C,I,X)

$$R = \begin{bmatrix} 8 & 5 & \boxed{3} & 5 & 9 & 1 & \boxed{3} & 2 & 7 & 9 & 3 \end{bmatrix}$$

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

$$I = \begin{bmatrix} 1 & 2 & 3 & 5 & 7 & 9 & 10 & 12 \end{bmatrix}$$

$$X = \begin{bmatrix} 1 & 2 & 3 & 5 & 7 & 9 & 10 & 12 \end{bmatrix}$$

$$R.*X(C)$$

$$R.*X(C) = \begin{bmatrix} 8 & 0 & 9 & 5 & 9 & 2 & 0 & 2 & 7 & 18 & 3 \end{bmatrix}$$

$$A \times x = \begin{bmatrix} 8 & 0 & 14 & 11 & 2 & 7 & 21 \end{bmatrix}$$

## Prodotto Matrice CRS e Vettore X

#### built-in function di Matlab utili:

# function cumsum(x) Somma cumulativa di un vettore

```
function diff(x)

ans =

1 3 7 7 10
```

#### Differenza tra gli elementi di x

N.B.: la dimensione di diff(x) è Inferiore di 1 rispetto a quella di x, a meno che non si aggiunga uno zi

a meno che non si aggiunga uno zero davanti

```
>> x = [1 2 4 0 3];
>> diff(x)
```

>> x = [1 2 4 0 3];

>> cumsum(x)

# Prodotto Matrice CRS e Vettore X

$$R.*X(C) = \begin{bmatrix} 8 & 0 & 9 & 5 & 9 & 2 & 0 & 2 & 7 & 18 & 3 \end{bmatrix}$$

$$cumsum \to \begin{bmatrix} 8 & 8 & 17 & 22 & 31 & 33 & 33 & 35 & 42 & 60 & 63 \end{bmatrix}$$

$$A \times x = \begin{bmatrix} 8 & 0 & 14 & 11 & 2 & 7 & 21 \end{bmatrix}$$

$$I = \begin{bmatrix} 1 & 2 & 3 & 5 & 7 & 9 & 10 & 12 \end{bmatrix}$$



Come legare cumsum, diff e il vettore I per ottenere il risultato?