Manuale base GNU/Octave

Nicola Ferru

11 giugno 2023

Indice

Capitolo 1

Introduzione

Definizione 1. *GNU/Octave è un applicativo per il calcolo matriciale che consente di svilgere tutte le operazioni base e non solo a riguardo, dallo somma, divisione, moltiplicazioni e sottrazioni tra matrici, calcolo del determinante, del grado e tanto altro.*

1.1 Pacchetti e impostazioni base

1.1.1 Pacchetti

Nome	**Descrizione**
fuzzy-logic-toolkit	Un toolkit di logica fuzzy per lo più compatibile con MATLAB per Octave
symbolic	Aggiunge funzionalità di calcolo simbolico a GNU Octave
Circuit Simulator (OCS)	Risolvere equazioni di circuiti elettrici DC e transitori.
Control	Strumenti CACSD (*Computer-Aided Control System Design*) per GNU Octave, basati sulla libreria SLICOT.
instrument-control	Funzioni I/O di basso livello per interfacce seriali, i2c, parallele, tcp, gpib, vxi11, udp e usbtmc.

Tabella 1.1: pacchetti utili

1.1.2 Impostazioni e formati

Nome	**Descrizione**
rat	aspetto rateo (invece dei numeri reali rende numeri frazionari)

Tabella 1.2: Impostazioni e formati

Capitolo 2

Funzioni base

2.1 Addizioni e sottrazioni tra matrici

$$A = \begin{vmatrix} 2 & 0 \\ 3 & -1 \end{vmatrix}, \; B = \begin{vmatrix} 4 & -1 \\ 1 & 2 \end{vmatrix} \in M_2(\mathbb{R}) \tag{2.1}$$

Calcolare $2A - 3B$ e $3A - 2B$, per svolgerlo non è complesso, infatti, il primo step è moltiplicare le matrici per il valore presente esternamente e poi fare la sottrazione tra matrici, il risultato è il seguente:

$$\begin{aligned} 2A - 3B = 2\begin{vmatrix} 2 & 0 \\ 3 & -1 \end{vmatrix} - 3\begin{vmatrix} 4 & -1 \\ 1 & 2 \end{vmatrix} &= \begin{vmatrix} 2 \cdot 2 & 2 \cdot 0 \\ 2 \cdot 3 & 2 \cdot -1 \end{vmatrix} + \begin{vmatrix} -3 \cdot 4 & -3 \cdot 1 \\ -3 \cdot 1 & -3 \cdot 2 \end{vmatrix} \\ &= \begin{vmatrix} 4 & 0 \\ 6 & -1 \end{vmatrix} + \begin{vmatrix} -12 & 3 \\ -3 & -6 \end{vmatrix} = \begin{vmatrix} -8 & 3 \\ 3 & -8 \end{vmatrix} \end{aligned}$$

stessa cosa ma con valori inversi

$$3A - 2B = \begin{vmatrix} -2 & 2 \\ 7 & -7 \end{vmatrix}$$

2.1.1 Soluzione per Octave o Mathlab

```
%% Prima operazione
A = [ 2, 0; 3, -1]; % Crea la prima matrice
B= [ 4, -1; 1, 2]; % Crea la seconda matrice
ris = 2*A-3*B; % svolge la prima operazione (2A-3B).
ris % stampa il risultato

%% seconda operazione
ris =3*A-2*B;
ris
```

Listing 2.1: svolgimento di una sottrazione tra matrici 2x2

Stampa a schermo

```
ris =

  -8   3
   3  -8

ris =

  -2   2
   7  -7
```

2.2 Determinante di una matrice

Un operazione molto utile è il determinante della matrice, fondamentale per lavorare su questa categoria di strutture, per calcolarlo non è difficile, in programmi come GNU/Octave e anche Matlab este la funzione **det**(M), che fa il classico svolgimento, prendendo un esempio concreto:

$$\begin{vmatrix} 3 & 5 \\ 8 & 4 \end{vmatrix} \tag{2.2}$$

Partendo da questa base dobbiamo fare la sequente operazione

$$\det(A) = \det \begin{vmatrix} 3 & 5 \\ 8 & 4 \end{vmatrix} = 3 \cdot 4 - 5 \cdot 8 = 12 - 40 = -28 \tag{2.3}$$

Quindi il determinante della matrice 2x2 A è -28, questo è anche il metodo che potete utilizzare su octave per fare la verifica del valore ottenuto con la funzione già pronta.

2.2.1 Soluzione per Octave o Mathlab

```
%% Svolgimenti interattivo
A=[3, 5; 8, 4];
ris = det(A);
A

ris

%% svolgimento manuale
ver =  A(1)*A(4) - A(3)*A(2);
ver
```

Listing 2.2: svolgimento del determinante di una matrice 2x2

Stampa a schermo

```
A =

   3   5
   8   4

ris = -28
ver = -28
```

2.3 Matrice inversa

Un operazione fondamentale è proprio la matrice inversa che serve per diverse formule presenti nel percorso di Ingegneria. quindi per calcolare l'inversa basta utilizzare il comando **inv**(M), uno dei problemi che si può riscontrare in questo caso è il fatto che il risultato possa essere espresso in numeri reali, cosa non molto pratica, quindi per sistemare questo problema basta applicare il formato rateo, come speficicato sopra, infatti, esiste una funziona di formato chiamata rat che può essere attivata con il semplice comando **format rat** e il problema verra risolto. Ma il metodo migliore è quello di fare un esempio. Prendiamo una matrice 3x3

$$A = \begin{vmatrix} 2 & 4 & 5 \\ 3 & 6 & 10 \\ 9 & 1 & 7 \end{vmatrix} \tag{2.4}$$

La sua inversa sarà A^{-1} che sarà composta dei seguenti valori

2.4 Diagonale di una matrice

Un altra funzione che in Matlab e octave viene fatta in modo pratico e veloce è la stampa della diagonale. Infatti, dentro l'ambiente viene utilizzato il comando **diag**(M).

$$A = \begin{pmatrix} 2 & 20 & 1 & 3 \\ 4 & 9 & 12 & 0 \\ 6 & 4 & 13 & 7 \\ 10 & 39 & 37 & 5 \end{pmatrix}$$

Questo comando va a creare un vettore composto da i numeri presenti nella diagonale della matrice, in questo caso l'istruzione **diag**(A), selezionera i numeri scritti in rosso:

$$A = \begin{pmatrix} \color{red}{2} & 20 & 1 & 3 \\ 4 & \color{red}{9} & 12 & 0 \\ 6 & 4 & \color{red}{13} & 7 \\ 10 & 39 & 37 & \color{red}{5} \end{pmatrix}$$

e quindi $ans = \begin{pmatrix} 2 & 9 & 13 & 5 \end{pmatrix}$, ovviamente questo accade nel caso base, perché il comando **diag** accetta al suo interno più di un parametri, infatti, se noi andiamo ad accodare al nominativo della matrice un numero possiamo ottenere le altre diagonali. Ed esempio se facciamo **diag**(A,1), il risultato sarà il seguente:

$$A = \begin{pmatrix} 2 & 20 & 1 & 3 \\ 4 & 9 & 12 & 0 \\ 6 & 4 & 13 & 7 \\ 10 & 39 & 37 & 5 \end{pmatrix}$$

Quindi il vettore risultante sarà composto nel seguente modo $ans = \begin{pmatrix} 20 & 12 & 7 \end{pmatrix}$ da questo si denota che il parametro che andiamo a passare serve semplicemente a distanziarsi positivamente o negativamente dalla diagonale 0, quella che divide la matrice in due perfettamente. Nel caso in cui venga passato un parametro negativo, in questo caso il -1, **diag**(A,−1) il risultato sarà il seguente:

$$A = \begin{pmatrix} 2 & 20 & 1 & 3 \\ 4 & 9 & 12 & 0 \\ 6 & 4 & 13 & 7 \\ 10 & 39 & 37 & 5 \end{pmatrix}$$

Nota Bene 1. *Il termine ans sta per Answer ed è il valore che viene salvato automaticamente dal calcolatore per rendelo, esso ha una funzione temporanea visto che verrà sovrascritto alla prossima operazione.*

2.4.1 Esempio in Matlab o Octave

```
A = [2, 20, 1, 3; 4, 9, 12, 0; 6, 4, 13, 7; 10, 39, 37, 5];

A
printf('________________________\n');

diagZ = diag(A); % diagonale 0
diagZ

diagU = diag(A,1); % diagonale successiva
diagU

diagMU = diag(A,-1); % diagonale precedente
diagMU
```

Listing 2.3: Esempio di utilizzo della funzione `diag()`

Stampa a schermo

```
A =

    2   20    1    3
    4    9   12    0
    6    4   13    7
   10   39   37    5

------------------------
diagZ =

    2
    9
   13
    5

diagU =

   20
   12
    7

diagMU =

    4
    4
   37
```

2.5 Operazioni tra vettori e mattrici

Un'altra operazione tipica è la somma tra un vettore "matrice unidimensionale" e una matrice NxM, quindi anche qui ci sono dei matodi grafici di svolgimento, ma partiamo dalle basi, prendiamo un vettore A e una matrici v

$$A = \begin{pmatrix} 2 & 3 & 4 \\ 1 & 5 & 7 \\ 9 & 8 & 6 \end{pmatrix}, \quad \vec{v} = \begin{pmatrix} 3 & 4 & 5 \end{pmatrix} \tag{2.5}$$

2.5.1 Addizioni e sottrazioni tra matrici

Addizioni

non modo non molto dissimile a quello che avveniva con la somma tra mattrici, anche nella somma tra un vettore e una Matrice si va assomare i membri dell'uno per quelli dell'altra, in questo caso nello specifico la prima colonna della matrice è stata moltiplicata per il primo elemento del vettore, la seconda colonna per il secondo elemento e così via.

$$A + \vec{v} = \begin{pmatrix} 3+2 & 4+3 & 5+4 \\ 3+1 & 4+5 & 5+7 \\ 3+9 & 4+8 & 5+6 \end{pmatrix} = \begin{pmatrix} 5 & 7 & 9 \\ 4 & 9 & 12 \\ 12 & 12 & 11 \end{pmatrix}$$

Sottrazioni

$$A - \vec{v} = \begin{pmatrix} 3-2 & 4-3 & 5-4 \\ 3-1 & 4-5 & 5-7 \\ 3-9 & 4-8 & 5-6 \end{pmatrix} = \begin{pmatrix} -1 & -1 & -1 \\ -2 & 1 & 2 \\ 6 & 4 & 1 \end{pmatrix}$$

2.5.2 Moltiplicazioni e divisioni

$$A \cdot \vec{v} = \begin{pmatrix} 3{\cdot}2 & 4{\cdot}3 & 5{\cdot}4 \\ 3{\cdot}1 & 4{\cdot}5 & 5{\cdot}7 \\ 3{\cdot}9 & 4{\cdot}8 & 5{\cdot}6 \end{pmatrix} = \begin{pmatrix} 6 & 12 & 20 \\ 3 & 20 & 35 \\ 27 & 32 & 30 \end{pmatrix}$$

In questo caso l'ambiguità non sta nell'operazione in se e per se ma nella sintassi di matlab e Octave che presentano due diverse funzioni per la moltiplicazione e la divisione, la prima è `A*M` che serve a fare una moltiplicazioni tra matrici della stessa dimensione e poi c'è quella per che forza la condizione `A.*M` che funziona con matrici di dimensione anche differente.