

***Il programma OCTAVE
per l'insegnamento dell'algebra lineare
nella Scuola Secondaria***

R. Vitolo

Dipartimento di Matematica – Università di Lecce

SaLUG! - Salento Linux User Group



OCTAVE è un programma per il calcolo matriciale (e non solo). Le sue caratteristiche sono:

- è a linea di comando;
- è *software libero* (e gratuito);
- è facilmente installabile;
- è semplice da apprendere.

NOTA: L'utilizzo di un programma non si apprende con un manuale, ma lavorando con il programma al calcolatore.

- *Come si ottiene aiuto su di un comando?* Basta scrivere al terminale
- *Come si introduce una matrice?* Basta scrivere al terminale la matrice:

```
> help nomecomando
```

```
> A=[1 2 3; 4 5 6; 7 8 9]
```

L'aggiunta di un ';' alla fine sopprime l'output del terminale. La matrice rimane in memoria, fino a che non viene cancellata o rimpiazzata, o il programma viene chiuso.

- *Come si seleziona un elemento di matrice? Con i suoi indici: A(2,3) stampa il valore 6.*
- *Come si inserisce una matrice complessa? Basta inserire i numeri complessi senza spazi così:*

```
> C=[1+i 2-2i; 3+i -0.5+i]
```

- *Come si ottiene la matrice trasposta? A'.*
- *Come si crea un vettore colonna? Così:*

```
> pippo=[1;2;3;4;5]
```

oppure così:

```
> pippo=[1 2 3 4 5]'
```

- *Come si ottengono le dimensioni di una matrice?*
> `[m, n] = size (pippo)`
- *Quali operazioni tra matrici sono disponibili?* $+$, $-$, $*$, \wedge , $'$, \backslash , $/$. Si inserisca un'altra matrice B di tipo e si calcoli $A+B$, $A-B$, $A*B$.
- *Che cos'è la "divisione matriciale"?* Equivale a risolvere un sistema lineare:
> `b = [1 2 3]'`
> `A = [1 2 1; -2 3 5; 8 0 2]`
> `X = A \ b`

Si effettui la verifica $A*X=b$.

- *Ci sono matrici predefinite? Si: **eye**(n), **zeros**(m,n), **rand**(m,n), **diag**, **triu** e **tril** , ecc..*
- *Come si crea una matrice di tipo 5×5 di numeri pseudocasuali compresi tra 0 e 100? Così:*

$$E = \text{floor} (100 * (\text{rand} (5 , 5)))$$

- *Come si seleziona una sottomatrice? Si consideri la sottomatrice di E formata dalle colonne 2 e 3 e dalle righe 3, 4, 5:*

$$E (2 : 3 , 3 : 5)$$

I : indicano un ciclo a passo 1. La prima riga si può ottenere da $E (1 , :)$.

- *Come si visualizzano le variabili?* **who**.
- *Come si salvano le variabili?* **save**.
- *Come si cancellano le variabili?* Così: **clear A** oppure **clear all**.
- *Come si ricaricano le variabili salvate?*
load nomefile.mat
- *Come si misura il tempo di calcolo?* Così:
> **t1=cputime**; **X=A\b**; **cputime-t1**
- *Come si seleziona il formato di output?* **format**
- *Qual'è la precisione del programma?* **eps**.

Problema 1. *Si moltiplichino le matrici A e B , dove A è la matrice quadrata che ha per diagonale il vettore $(1, 2, -3)$ e*

$$B = \begin{pmatrix} 1 & 2 \\ -1 & -1 \\ 0 & 4 \end{pmatrix}.$$

SOLUZIONE. > **sol=diag** ([1 2 -3])*B

Problema 2. *Si verifichi che il determinante di*

$$A = \begin{pmatrix} 1 & 3 & -2 & -2 \\ 0 & 2 & 4 & -5 \\ -1 & 2 & 3 & 2 \\ 0 & -1 & -9 & 3 \end{pmatrix}$$

*calcolato con la regola di Laplace è uguale al determinante di A calcolato con la funzione **det**.*

SOLUZIONE. Si ha **det**(A)=166. Si usa la quarta riga per sviluppare **det**(A): si ottiene

$$\begin{aligned} > \text{temp} = 1 * \mathbf{det}(A(1:3, [1 \ 3 \ 4])) \\ & - 9 * \mathbf{det}(A(1:3, [1 \ 2 \ 4])) - 3 * \mathbf{det}(A(1:3, 1:3)) \end{aligned}$$

Problema 3. *Si verifichi che il determinante di A cambia segno se si scambiano due righe.*

SOLUZIONE. Si utilizzi l'assegnazione

```
> temp=A([2 1 3 4],:)
```

per effettuare lo scambio di righe.

Problema 4. *Si verifichi che il determinante di A è moltiplicato per 5 se si moltiplica la prima riga per 5. Che succede se si moltiplica A per 5?*

Problema 5. *Data la matrice*

$$B = \begin{pmatrix} 2 & 0 & 4 & -10 \\ 1 & -2 & -4 & 0 \\ 6 & -2 & -2 & 1 \\ 3 & 2 & -4 & -9 \end{pmatrix},$$

si verifichi la regola di Binet per il prodotto AB .

Problema 6. *Si verifichi che $AB \neq BA$, $(AB)^T \neq A^T B^T$ ma che $(AB)^T = B^T A^T$.*

Problema 7. Si calcoli la matrice inversa della matrice

$$D = \begin{pmatrix} 1 & 2 & 0 \\ 2 & 1 & 0 \\ 6 & 6 & 1 \end{pmatrix}$$

mediante la matrice dei complementi algebrici.

SOLUZIONE. Si calcola solo il primo complemento algebrico

$$> D_{11} = (-1)^{(1+1)} \mathbf{det} (D(2:3, 2:3))$$

L'inversa sarà data da

$$> E = (\mathbf{det} (D))^{-1} * ([D_{11} \ D_{12} \ D_{13}; \ D_{21} \ \dots \]')$$

Si verifichi che questa coincide con la matrice $\mathbf{inv}(D)$.

Problema 8. *Si calcoli il rango della matrice*

$$F = \begin{pmatrix} 1 & 2 & 0 & 1 & 1 \\ 1 & 2 & 1 & 0 & 2 \\ 2 & 4 & 1 & 1 & 3 \\ 3 & 6 & 2 & 1 & 5 \end{pmatrix}$$

utilizzando il teorema degli orlati (o di Kronecker).

SOLUZIONE. Si trovino minori di F di dimensioni crescenti con determinante diverso da 0. Si calcolino gli orlati di questi per stabilire il rango. Si confronti la soluzione ottenuta con quella che si ha dal comando **rank**.

Problema 9. *Date le matrici*

$$G = \begin{pmatrix} 1 & 2 \\ -3 & 5 \end{pmatrix}, \quad H = \begin{pmatrix} 8 & 6 \\ 2 & -7 \end{pmatrix}$$

provare che $(A + B)^2 \neq A^2 + 2AB + B^2$.

È possibile automatizzare le operazioni in OCTAVE in due modi: tramite i *function files* (o funzioni). Una funzione

- è scritta in un file di testo (anche col 'blocco note');
- inizia con la riga **function** xyz dove 'xyz.m' sarà il nome del file;
- è *interpretata* come il BASIC, non *compilata* come il FORTRAN o il c.

Esempio 1. *Scrivere una funzione che stampi la frase `ciao mondo` sullo schermo.*

SOLUZIONE.

```
function ciao
    printf ( "ciao _mondo\n" );
endfunction
```

L'istruzione `endfunction` non è strettamente necessaria.

Esempio 2. *Scrivere una funzione che calcoli il fattoriale di un numero n dato.*

SOLUZIONE.

```
function valore = fattoriale (n)
  if (n > 0)
    valore = n * fattoriale (n-1);
  else
    valore = 1;
  end
endfunction
```

Si noti che la funzione è stata definita ricorsivamente.

Esempio 3. *Scrivere una funzione che calcoli la media aritmetica di un vettore v dato.*

SOLUZIONE.

```
function retval = media (v)
    retval = sum (v) / length (v);
endfunction
```

Esempio 4. *Scrivere una funzione che calcoli la media aritmetica di un vettore v dato, effettuando prima un controllo sul tipo di dato.*

SOLUZIONE.

```
function retval = mediacontr (v)
    retval = 0;
    if (is_vector (v))
        retval = sum (v) / length (v);
    else
        error ("mediacontr: _dato _non _vettoriale");
    end
endfunction
```

Esempio 5. *Scrivere una funzione (di due variabili) che calcoli l'area di un rettangolo.*

SOLUZIONE.

function

```
[ A , p , d ] = rettang ( a , b )
```

```
  A = a * b;
```

```
  p = 2 * ( a + b );
```

```
  d = sqrt ( a2 + b2 );
```

endfunction

Esempio 6. *Scrivere una funzione che crei una matrice triangolare superiore di numeri casuali compresi tra 0 e 100 di ordine n .*

SOLUZIONE.

```
function a=mat_sup(n)
    temporanea=floor(100*rand(n,n))
    for j=1:n
        for i=1:j
            a(i,j)=temporanea(i,j);
        end
    end
endfunction
```

Esempio 7. *Scrivere una funzione di due variabili che risolva un sistema con matrice triangolare superiore. Iniziare la funzione con un test sulle variabili.*

SOLUZIONE.

```
function x=tri_sup(a,b)
% Controlla le dimensioni di a:
[n,m] = size(a);
if m ~ = n
    printf( 'A non e' ' quadrata ' )
    return
end
```

```
for i=1:n
  for j=1:i-1
    if a(i,j) ~=0
      printf( 'A_non_e' '_triangolare_supe
    return
    end
  end
end
if length(b) ~= n
  printf( 'B_non_e' '_compatibile' )
  return
end
```

```
% Risolve il sistema
x(n) = b(n)/a(n,n);
for i=n-1:-1:1
    sum=b(i);
    for j=i+1:n
        sum = sum - a(i,j)*x(j);
    end
    x(i) = sum/a(i,i);
end
endfunction
```

Problema 10. *Scrivere una funzione che crei una matrice di ordine n che ha 2 sulla diagonale principale, -1 sulla sottodiagonale e sulla sopradiagonale e 0 altrimenti.*

Problema 11. *Scrivere un programma che confronti la velocità di risoluzione di un sistema triangolare superiore usando, rispettivamente, la funzione dell'esercizio precedente e la funzione di 'divisione matriciale' di OCTAVE.*

Problema 12. *Scrivere una funzione di due variabili (matrice incompleta e termine noto) che risolva un sistema con matrice triangolare inferiore. Iniziare la funzione con un test sulle variabili.*

Problema 13. *Scrivere una funzione di una variabile (matrice quadrata) che calcoli il determinante con uno sviluppo di Laplace. Si confronti il tempo di esecuzione con il tempo di esecuzione della funzione 'det' e con la funzione dell'esercizio precedente.*

Problema 14. *Scrivere una funzione di una variabile che calcoli l'inversa di una matrice quadrata con il metodo di Cramer–Laplace. Si confronti il tempo di esecuzione con il tempo di esecuzione della funzione 'inv'.*

Problema 15. *Scrivere una funzione di due variabili (matrice incompleta e termine noto) che risolva un sistema lineare a matrice quadrata con il metodo di eliminazione. Alla fine, si usi la funzione 'tri_sup'.*