Esercizi di MatLab

Sommario

Esercizi di introduzione a MatLab per il corso di Analisi Numerica I, A.A. 2017–2018.

Gli esercizi sono divisi in due gruppi: fondamentali ed avanzati. I primi sono esercizi basilari per prendere familiarità con l'ambiente MatLab, i secondi richiedono una conoscenza leggermente più ampia.

Indice

1 Esercizi fondamentali: calcolo vettoriale e matriciale

1. Generare i vettori $v = (1, 2, \dots, 6)^\top$ e $w = (1, 2, \dots, 6)^\top$ (vettori colonna) ed effettuare le seguenti operazioni:
   (a) `a = v+w`;
   (b) `b = 4v`;
   (c) calcolare il prodotto di `v` per `w` elemento per elemento e salvare il risultato nella variabile `c`;
   (d) dividere ogni elemento di `w` per due e salvare il risultato nella variabile `d`;
   (e) dividere ogni elemento di `v` per il corrispondente elemento di `w` e salvare il risultato nella variabile `e`;
   (f) calcolare il prodotto scalare di `v` e di `w`, salvare il risultato in `f`;
   (g) calcolare `g = 2v-6w`;
   (h) memorizzare negli elementi di posto pari del vettore `g1` gli elementi di posto pari di `d` e negli elementi di posto dispari di `g1` gli elementi di posto dispari di `c`;
   (i) creare il vettore `h` con 5 copie del vettore `v`;
   (j) osservare il comportamento del comando `h([6:6:end])`;
   (k) sostituire 0 negli elementi con indice multiplo di 5 in `h` e 1 negli elementi con indice multiplo di 6;
   (l) dato $u = (1, 2, \dots, 6)$ (vettore riga) calcolare `u*w` e `w*u`: osservare i risultati e capire cosa succede.
2. Generare una matrice quadrata $A \in \mathbb{M}_n(\mathbb{R})$ (ove n è un dato introdotto dall'esterno) con elementi generati a caso da una distribuzione uniforme nell'intervallo $[0, 1]$ e calcolare (eventualmente usando le funzioni cumulative):
   (a) un vettore che contiene la somma degli elementi di ciascuna colonna;
   (b) un vettore che contiene la somma degli elementi di ciascuna riga;
   (c) un vettore che contiene la somma degli elementi al quadrato di ciascuna riga;
   (d) il massimo degli elementi della matrice;
   (e) la somma di tutti gli elementi della matrice.
3. Generare i vettori colonna `v,w` $\in \mathbb{R}^{25}$ da una distribuzione uniforme nell'intervallo $[0, 1]$, Effettuare i seguenti passaggi:
   (a) `t` $= \langle$`v,w`$\rangle$`v` $+ \langle$`v,w`$\rangle$`w`;
   (b) dato `s` $= (1, \dots, 25)^\top$, memorizzare in `p` la divisione elemento per elemento di `t` per `s`;
   (c) data la matrice $A \in \mathcal{M}_{3\times 25}(\mathbb{R})$, generata da una distribuzione uniforme in $[0, 1]$, salvare in `q` il prodotto `Ap`;
   (d) creare la matrice identità `I` di ordine 3, salvare in `e1` la prima colonna di `I`, in `e2` la seconda colonna e in `e3` la terza colonna. Salvare nella variabile `a1` il prodotto scalare $\langle$`q,e_1`$\rangle$, in `a_2` il prodotto scalare $\langle$`q,e_2`$\rangle$ e in `a_3` il prodotto scalare $\langle$`q,e_3`$\rangle$;
   (e) creare il vettore `a` $=$ `(a_1,a_2,a_3)`$^\top$ e verificare che sia identico al vettore `q`;
   (f) memorizzare in un vettore `b` gli elementi di posto pari di `t`, moltiplicarli per 10 e arrotondarli utilizzando il comando `fix`;
   (g) creare il vettore `c = (1,2,3)`$^\top$, memorizzare in `D` il prodotto `c*b`$^\top$: descrivere l'output ottenuto;
   (h) data la matrice `B` costituita da 3 copie del vettore `q`, effettuare le operazioni `BA` e `AB`: sono consentite? Dare una motivazione in entrambi i casi, negativo e affermativo.
4. Date le matrici `A=[1 2 3; 4 5 6; 7 8 9], B=[2 -1 0; -1 2 -1; 0 -1 2]`, descrivere gli output delle seguenti istruzioni:
   (a) `A(:,[1,3])=B(:, 1:2);`
   (b) `C=A./B;`
   (c) `C=A.^B;`
   (d) `C=triu(A)+tril(B,-1);`
   (e) `A([1:2],:)=[];`

(f) `D=B([3,2],1:2:3);`

5. Creare una matrice $A \in \mathcal{M}_5(\mathbb{R})$ a piacere ed effettuare le seguenti operazioni:
   (a) memorizzare in `v` la sua seconda riga;
   (b) memorizzare in `w` la sua terza colonna;
   (c) estrarre la sottomatrice $B \in \mathcal{M}_3(\mathbb{R})$ a parte dall'elemento `A(2,2)`;
   (d) creare una matrice $C \in \mathcal{M}_7(\mathbb{R})$ con la prima e l'ultima colonna di zeri e la prima e ultima riga di 1, e al "centro" porre la matrice A;
   (e) creare la matrice $D \in \mathcal{M}_3(\mathbb{R})$ in cui ogni elemento è il prodotto degli indici di riga e colonna: verificare se è simmetrica;
   (f) eseguire `BD` e `DB`: sono diversi?
   (g) eseguire il prodotto componente per componente di `B` e `D` e vedere se questo corrisponde a `B*D` o `D*B`;
   (h) memorizzare in `E` la combinazione `B*D^2-3*B*D-7*D^2` ;
   (i) verificare che `D'*B'=(B*D)'`;
   (j) calcolare `Dw1`, dove `w_1` è il vettore contenente i primi tre elementi di `w`;
   (k) calcolare `v1D`, dove `v1` è il vettore contenente i primi tre elementi di `v`;
   (l) creare la seguente matrice a blocchi:
   $$\begin{pmatrix} I_r & B & 0_r \\ B & D & -B \\ 0_r & -B & -I_r \end{pmatrix}$$
   con r opportuno. I_r è la matrice identità di ordine r, mentre 0_r è la matrice quadrata di dimensione r con ogni elemento pari a 0.

6. Dato un vettore $x = (x_1, x_2, \dots, x_n)^\top$ e un naturale $m > 0$, costruire la seguente matrice:
   $$V = \begin{pmatrix} 1 & x_1 & x_1^2 & x_1^3 & \dots & x_1^{m-1} \\ 1 & x_2 & x_2^2 & x_2^3 & \dots & x_2^{m-1} \\ \vdots & & & & & \vdots \\ 1 & x_n & x_n^2 & x_n^3 & \dots & x_n^{m-1} \end{pmatrix}$$
   utilizzando la sintassi vettoriale di MatLab. Successivamente, creare la stessa matrice usando due cicli `for` innestati. Misurare il tempo necessario per la costruzione di tale matrice con i due metodi utilizzando i comandi `tic` e `toc`.

7. Quali elementi contiene il vettore `z` dopo i ciascuno dei seguenti comandi di Matlab?
   (a) `z=[10 40 20 80 30 70 60 90];`
   `z(1:2:7)=zeros(1,4);`
   (b) `z=[10 40 20 80 30 70 60 90];`
   `z(7:-2:1)=zeros(1,4);`
   (c) `z=[10 40 20 80 30 70 60 90];`
   `z([3 4 8 1])=zeros(1,4);`

8. Data la matrice `A=[2 2 1; 1 -1 4; 2 1 -3]`, descrivere l'output dei seguenti comandi Matlab:
   (a) `[p,q]=size(A(1:2,1:2));`
   (b) `A./A';`
   (c) `A(1,:)=A(2,:).*A(3,:);`
   (d) `A(2,:)=A(:,1)';`
   (e) `A(:,2)=A(:,2).^3;`

9. Creare due matrici quadrate `A,B` di dimensione $50 < n < 1000$ e un vettore `v` di dimensioni opportune. Utilizzando i comandi `tic` e `toc`, verificare la differenza di tempo di calcolo tra `A*B*v` e `A*(B*v)`.

10. Data la matrice $A \in \mathcal{M}_n(\mathbb{R})$, n scelto da tastiera, salvare in v la sua vettorizzazione tramite l'uso del comando jolly `:`. Calcolare `V=v.^2`, eseguire il comando `B = reshape(V,n,n)` e controllare che `B==A.^2`.

11. Creare un vettore `x` di 20 elementi i cui elementi siano equispaziati fra 0 e 2π. Salvare in `y` i valori della funzione seno calcolata in `x`, salvare in `z` i valori della funzione coseno calcolata in `x`. Utilizzando il comando `fprintf`, stampare a video una tabella di tre colonne in cui sulla prima colonna devono essere presenti i valori di `x`, sulla seconda i valori di `Y` e infine sulla terza i valori di `z`.

12. Dati `A=[1 2 3; 4 5 6]; B=[2 2 1;1 0 3]; x=[1 2 3];y=[4 5 6];` descrivere gli output delle seguenti istruzioni:
    (a) `C=A.*B;`;
    (b) `z=A(1,:).*y`;
    (c) `z=x./ [B(:,2);3];`;
    (d) `z=x.\y`;
    (e) `z=A(2,:).^ B(1,3:-1:1);`.

13. Dati i vettori `xs = ones(10,1)` e `y = 2*randn(10,1)`, descrivere gli output dei seguenti comandi senza eseguirli:
    (a) `abs(xs-y)`;
    (b) `norm(xs-y)/norm(xs)`;
    (c) `abs(xs-y)./abs(xs)`.

    Verificare i risultati al calcolatore.

14. Data una matrice $A \in \mathcal{M}_n(\mathbb{R})$, $2 < n < 10$, contentente numeri random compresi fra -10 e 10 (utilizzare la funzione `rand`) effettuare i seguenti passaggi:
    (a) estrarre in `d` la diagonale di `A`;
    (b) salvare in `B` la parte triangolare superiore di `A`;
    (c) sottrarre a `B` la diagonale di `A` (hint: controllare l'uso della funzione `triu` per fare tutto in un passaggio);
    (d) porre `C = B+B'`;
    (e) dato `v` di dimensioni opportune, verificare che `v'*C*v` sia maggiore di zero o meno, stampando un messaggio a video, mediante l'utilizzo combinato della funzione `fprintf` e del ciclo di controllo `if-then-else`;
    (f) salvare in `D` la sottomatrice di `A` costituita dalle prime tre colonne e dalle ultime due righe di `A` ; scrivere in `E` il prodotto `D'*D`;
    (g) dato `v` di dimensioni opportune, verificare che `v'*E*v` sia maggiore di zero o meno, stampando un messaggio a video, mediante l'utilizzo combinato della funzione `fprintf` e del ciclo di controllo `if-then-else`;
    (h) verificare che `E` sia simmetrica.

2 Esercizi fondamentali: function

Consiglio: scrivere un `main` che chiami ogni funzione dei successivi esercizi, invece di chiamare le funzioni da shell di comando. Per esempio, se per il primo esercizio si utilizza la funzione `my_factorial`, allora il `main` che chiamerà tale funzione potrebbe essere

```
clear all
close all
clc

n = input('Scegliere un numero\n');      % potete decidere se introdurlo
                                         % da tastiera o definirlo nel main

f = my_factorial(n);
```

1. Scrivere una function che preso in input un numero $0 \leq n \leq 10$ naturale, ne calcoli il fattoriale.
2. Scrivere una function che presi in input un vettore x e un numero intero n, restituisca il vettore y le cui componenti sono
$$y_i = \sum_{k=0}^{n} \frac{x_i^k}{k!}$$
Confrontare il risultato con la funzione nativa `exp` al variare di n.
3. Scrivere una function che, date in input le coordinate di un punto $x = (x_1, x_2) \in \mathbb{R}^2$ e un raggio R, controlli che tale punto sia all'interno o meno della circonferenza di raggio R, stampando un messaggio a video; la function dovrà restituire anche la lunghezza di tale vettore.
4. Scrivere una function che, dato in input una matrice $A \in \mathcal{M}_2(\mathbb{R})$, restituisca il suo determinante. Confrontare il risultato ottenuto con il risultato fornito dalla funzione nativa `det`.
5. Scrivere una function che, data in input una matrice $A \in \mathcal{M}_n(\mathbb{R})$ con n a piacere, restituisca la diagonale di A, il valore massimo e il suo valore minimo di A.
6. Scrivere una function che, dati in input tre vettori $v \in \mathbb{R}^n, w \in \mathbb{R}^{n-1}, t \in \mathbb{R}^{n-1}$ restituisca una matrice la cui diagonale è costituita dal vettore v, la prima sovradiagonale dal vettore w e la prima sottodiagonale dal vettore t.
7. Data la matrice
$$A = \begin{pmatrix} 8 & 9 & 2 \\ 9 & 6 & 5 \\ 1 & 0 & 9 \end{pmatrix}$$
e il vettore $b = (19, 20, 10)^\top$, scrivere una function che, preso in input un vettore y, controlli che esso sia soluzione del sistema $Ax = b$.
8. Dato un polinomio p, scrivere una function che, dati in input un punto x_0 e i coefficienti del polinomio, calcoli il valore del polinomio in x_0. Confrontare il risultato ottenuto con il risultato fornito dalla funzione nativa `polyval`. Hint: i coefficienti del polinomio possono essere memorizzati in un vettore. Per esempio, i coefficienti di
$$p(x) = 4x^3 + x^2 - 1$$
possono essere memorizzati nel vettore `p = [4 1 0 -1]`.

9. Data una matrice $A \in \mathcal{M}_n(\mathbb{R})$ con n a piacere, scrivere una function che, data in input A, restituisca le tre matrici L, M, N tali che

$$L = \begin{pmatrix} a_{11} & 0 & \dots & 0 & 0 \\ 0 & a_{22} & 0 & \dots & 0 \\ \vdots & & \ddots & & \vdots \\ \vdots & & & a_{n-1n-1} & 0 \\ 0 & 0 & \dots & 0 & a_{nn} \end{pmatrix}$$

$$M = \begin{pmatrix} 0 & 0 & \dots & 0 & 0 \\ -a_{21} & 0 & 0 & \dots & 0 \\ -a_{31} & -a_{32} & 0 & \dots & 0 \\ \vdots & & \ddots & & \vdots \\ -a_{n1} & -a_{n2} & \dots & -a_{nn-1} & 0 \end{pmatrix}, N = \begin{pmatrix} 0 & -a_{12} & -a_{13} & \dots & -a_{1n} \\ 0 & 0 & -a_{23} & \dots & -a_{2n} \\ 0 & 0 & 0 & \dots & -a_{3n} \\ \vdots & & \ddots & & \vdots \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

Utilizzare i comandi `tril` e `triu`.

10. Scrivere una function che, dato in input un naturale n, restituisca in output il valore

$$\left(1 + \frac{1}{n}\right)^n$$

e la differenza fra il valore ottenuto e il numero e; la function dovrà stampare un messaggio a video contenente tale differenza con almeno 8 cifre significative (consultare l'help di `fprintf`).