Risoluzione di equazioni in MATLAB

Eugenio G. Omodeo

OLTRE L'ALGEBRICO! *Dal* [DAVE HESLOP]:

Figure 3.2: *How can we find the true count rate of a Geiger counter which suffers from dead-time.*

Trieste, dopo 27.10.2021

Punti che verranno trattati:

1. risoluzione di sistemi *lineari* di equazioni
2. risoluzione di equazioni polinomiali mono-var.
3. soluzione per tentativi di un sistema di equazioni diofantee
4. sol. per tentativi dell'eq. trascendente di $\boxed{\boxed{\text{contatore Geiger}}}$

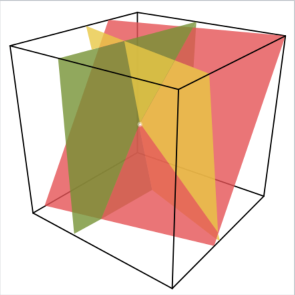
In mathematics, a **system of linear equations** (or **linear system**) is a collection of one or more linear equations involving the same set of variables.[1][2][3][4][5] For example,

$$\begin{alignedat}{4} 3x &\; + \;& 2y &\; - \;& z &\; = \;& 1 \\ 2x &\; - \;& 2y &\; + \;& 4z &\; = \;& -2 \\ -x &\; + \;& \tfrac{1}{2}y &\; - \;& z &\; = \;& 0 \end{alignedat}$$

is a system of three equations in the three variables x, y, z. A **solution** to a linear system is an assignment of values to the variables such that all the equations are simultaneously satisfied. A solution to the system above is given by

$$\begin{aligned} x &= 1 \\ y &= -2 \\ z &= -2 \end{aligned}$$

since it makes all three equations valid. The word "system" indicates that the equations are to be considered collectively, rather than individually.



A linear system in three variables determines a collection of planes. The intersection point is the solution.

Il sistema visto sopra, ad es., può essere pensato cosí:

$$\begin{bmatrix} 3 & 2 & -1 \\ 2 & -2 & 4 \\ -1 & \frac{1}{2} & -1 \end{bmatrix} * \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} 1 \\ -2 \\ 0 \end{bmatrix};$$

o anche, passando alle trasposte:

$$\begin{bmatrix} x & y & z \end{bmatrix} * \begin{bmatrix} 3 & 2 & -1 \\ 2 & -2 & 4 \\ -1 & \frac{1}{2} & -1 \end{bmatrix}' = \begin{bmatrix} 1 & -2 & 0 \end{bmatrix}.$$

Anyway, the best way to think about all matrix division is in terms of solving linear systems. MATLAB interprets

```
>> x = A/B
```

as **"solve the linear system x*B = A (for x)"**. And, similarly,

```
>> x = A\B
```

is **"solve the linear system A*x = B (for x)"**. MATLAB will solve the system if at all possible (ie if the dimensions are consistent), giving, in general, the least-squares solution (ie minimizing the 2-norm of the residual). This means it will "solve" over/under/determined systems, in the most natural way possible -- the actual solution if there is one, or the least-squares solution otherwise.

Command Window

New to MATLAB? See resources for Getting Started.

```
>> X = [ 3   2   -1
         2  -2    4
        -1 0.5   -1 ] \ [ 1 ; -2 ; 0 ]

X =

    1.0000
   -2.0000
   -2.0000

>> X = [ 1  -2   0 ] / [ 3   2   -1
                         2  -2  0.5
                        -1   4   -1 ]

X =

    1.0000   -2.0000   -2.0000

fx >>
```

Cosa non va in questi due sistemi lineari?

```
>> [3 2 4; -1 1 2; 9 5 10]\[0;0;0]
Warning: Matrix is close to singular or badly
scaled. Results may be inaccurate. RCOND =
8.077019e-19.


ans =

     0
     0
     0

>> [3 2 4; -1 1 2; 9 5 10]\[1;2;3]
Warning: Matrix is close to singular or badly
scaled. Results may be inaccurate. RCOND =
8.077019e-19.


ans =

   1.0e+16 *

  -0.000000000000000
   2.579334332039467
  -1.289667166019733

>> rank([3 2 4; -1 1 2; 9 5 10])

ans =

     2
```

☞ ☞

La matrice

$$\begin{pmatrix} 3 & 2 & 4 \\ -1 & 1 & 2 \\ 9 & 5 & 10 \end{pmatrix}$$

dei due sistemi ha una colonna che è multipla di un'altra: essa ha dunque $\boxed{\textit{rango}}$ 2 *e* $\boxed{\textit{determinante}}$ 0 *!!*

Ne consegue che il primo sistema è sottodeterminato.

E l'altro ?

☞ ☞

La matrice

$$\begin{pmatrix} 3 & 2 & 4 \\ -1 & 1 & 2 \\ 9 & 5 & 10 \end{pmatrix}$$

dei due sistemi ha una colonna che è multipla di un'altra: essa ha dunque $\boxed{\boxed{\textit{rango}}}$ *2* e $\boxed{\boxed{\textit{determinante}}}$ *0 !!*

Ne consegue che il primo sistema è sottodeterminato.

E l'altro ?

Esercizio:

Trovare—usando MATLAB—un k e un ℓ tali che

$$k \cdot (3 \;\; 2 \;\; 4) + \ell \cdot (-1 \;\; 1 \;\; 2) = (9 \;\; 5 \;\; 10)$$

Domanda

Che risposta vi aspettate da

```
>> A=[3 2 -1; 2 -2 (1/2); -1 4 -1]; B=[1;-2;0];
>> A\B == [det([B,A(:,2:3)])/det(A);
           det([A(:,1),B,A(:,3)])/det(A);
           det([A(:,1:2),B])/det(A)]
```

?

Domanda sul metodo di Cramer

Che risposta vi aspettate da

```
>> A=[3 2 -1; 2 -2 (1/2); -1 4 -1]; B=[1;-2;0];
>> A\B == [det([B,A(:,2:3)])/det(A);
           det([A(:,1),B,A(:,3)])/det(A);
           det([A(:,1:2),B])/det(A)]
```

```
ans =

  3×1 logical array

   1
   0
   0
```

?

Domanda sul metodo di Cramer

Che risposta vi aspettate da

```
>> A=[3 2 -1; 2 -2 (1/2); -1 4 -1]; B=[1;-2;0];
>> A\B == [det([B,A(:,2:3)])/det(A);
           det([A(:,1),B,A(:,3)])/det(A);
           det([A(:,1:2),B])/det(A)]
```

```
ans =

  3×1 logical array

   1
   0
   0
```

?

```
>> 10^-14 > abs(A\B - [det([B,A(:,2:3)])/det(A);
                       det([A(:,1),B,A(:,3)])/det(A);
                       det([A(:,1:2),B])/det(A)])

ans =

  3×1 logical array

   1
   1
   1
```

Live Editor – EqSecondoGrado.mlx *

+7 | crackSafe.m | digitClose.m | crackWeakSafe.m | classifyingSediments.m | nsceltek.m | daOttimizzare.m | bidimDaOttimizzare.m

Risolvo equazioni di secondo grado, di cui una a coefficienti letterali

```
syms x a b c

solve( 3*x^2 - 4*x + 1 == 0, x )

solve( 3*x^2 - 2*x + 1, x )

eqn = a*x^2 + b*x + c == 0

solve( eqn, x )
```

$$\begin{pmatrix} \frac{1}{3} \\ 1 \end{pmatrix}$$

ans =

$$\begin{pmatrix} \frac{1}{3} - \frac{\sqrt{2}\,\mathrm{i}}{3} \\ \frac{1}{3} + \frac{\sqrt{2}\,\mathrm{i}}{3} \end{pmatrix}$$

eqn = $a\,x^2 + b\,x + c = 0$

ans =

$$\begin{pmatrix} -\frac{b + \sqrt{b^2 - 4\,a\,c}}{2\,a} \\ -\frac{b - \sqrt{b^2 - 4\,a\,c}}{2\,a} \end{pmatrix}$$

Domanda (anche oltre il 2º grado):

Che differenze ci sono fra le funzioni `solve` e `roots` di MATLAB ?

Esercizio: Risolvere tramite MATLAB le seguenti equazioni di secondo e terzo grado nell'incognita x:

$$
\begin{array}{rcrcrcrcl}
 & & -2 \cdot x^2 & + & 10 \cdot x & - & 12 & = & 0\,, \\
-2 \cdot x^3 & + & 18 \cdot x^2 & - & 52 \cdot x & + & 48 & = & 0\,, \\
x^3 & - & 0.731 \cdot x^2 & - & 3.64 \cdot x & - & 125.92 & = & 0\,, \\
a \cdot x^3 & + & b \cdot x^2 & + & c \cdot x & + & d & = & 0\,.
\end{array}
$$

Esercizio: Prodotto di polinomi monovariati

Calcolate la lista dei coefficienti di ciascuno dei polinomi:

$$
\begin{array}{rcrcrcrcr}
 & & & & & & -2 & \cdot & (x-2)\,, \\
 & & & & -2 & \cdot & (x-2) & \cdot & (x-3)\,, \\
 & & -2 & \cdot & (x-2) & \cdot & (x-3) & \cdot & (x-4)\,, \\
-2 & \cdot & (x-2) & \cdot & (x-3) & \cdot & (x-4) & \cdot & (x+1)\,,
\end{array}
$$

e dell'ultimo polinomio ottenuto trovate con MATLAB le radici.

Risolviamo per tentativi questo sistema:

$$\begin{cases} 6\,w + 2\,x^2 - y^3 & = & 0\,, \\ 5\,x\,y - z^2 - 1 & = & 0\,, \\ w^2 - w + 2\,x - y + z - 3 & = & 0\,. \end{cases}$$

Risolviamo per tentativi questo sistema:

$$\begin{cases} 6\,w + 2\,x^2 - y^3 &= 0\,, \\ 5\,x\,y - z^2 - 1 &= 0\,, \\ w^2 - w + 2\,x - y + z - 3 &= 0\,. \end{cases}$$

```
cen = 100; % Da cambiare per modificare l'intervallo di ricerca

for w=-cen:1:cen, for x=-cen:1:cen,
    for y=-2*cen:2:2*cen, for z=-2*cen-1:2:2*cen+1,
        if (6*w+2*x^2-y^3==0  &
            5*x*y-z^2-1==0      &
            w^2-w+2*x-y+z-3==0)
                                    w, x, y, z,
end; end; end; end; end;
```

Risolviamo per tentativi questo sistema:

$$\begin{cases} 6\,w + 2\,x^2 - y^3 & = & 0\,, \\ 5\,x\,y - z^2 - 1 & = & 0\,, \\ w^2 - w + 2\,x - y + z - 3 & = & 0\,. \end{cases}$$

Esercizio: REIMPLEMENTAZIONE 'A SPIRALE'

Riprogettare la ricerca della soluzione impiegando while invece di for e facendola procedere da valori 'piccoli' (in valore assoluto) per le incognite verso valori grandi.

Risolviamo per tentativi questo sistema:

$$\begin{cases} 6\,w + 2\,x^2 - y^3 & = & 0\,, \\ 5\,x\,y - z^2 - 1 & = & 0\,, \\ w^2 - w + 2\,x - y + z - 3 & = & 0\,. \end{cases}$$

Esercizio: Reimplementazione 'a spirale'

Riprogettare la ricerca della soluzione impiegando while invece di for e facendola procedere da valori 'piccoli' (in valore assoluto) per le incognite verso valori grandi.

Domanda:

Pensate che la ricerca possa portare a un risultato se a posto del -3 mettiamo -4 ?

Funzionamento [modifica | modifica wikitesto]

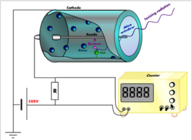
Il contatore Geiger è una camera a deriva utilizzata nel limite in cui la tensione satura, ovvero in modo che la tensione prodotta dal passaggio della particella ionizzante non dipenda dall'energia rilasciata da questa e quindi dal numero delle coppie ione-ione prodotte.

Quando una radiazione attraversa il tubo e colpisce una molecola del gas, la ionizza creando una coppia ione-elettrone. In questi dispositivi la carica raccolta è indipendente dalla ionizzazione primaria: come nelle altre camere a deriva gli ioni primari vengono accelerati a sufficienza da creare ionizzazioni secondarie quando urtano altre molecole di gas, ma la peculiarità del contatore geiger è che il campo elettrico è talmente intenso che anche le ionizzazioni secondarie creano a loro volta ulteriori ionizzazioni[1][2]. Questo processo è detto moltiplicazione a valanga.

L'impulso elettrico risultante sarà testimone dell'avvenuto contatto con una radiazione ionizzante e sarà contato da un circuito elettronico, i famosi "click" che si sentono. A seconda del numero di conteggi fatti in un'unità di tempo si riesce a capire se si è in presenza di una sorgente radioattiva e la sua pericolosità.

Il contatore Geiger non effettua una misura operativa della grandezza esposizione/kerma in aria, ma si limita a mettere in relazione il numero di conteggi con la grandezza dosimetrica. Per questo la sensibilità dello strumento varia significativamente al variare dell'energia della radiazione incidente. L'effetto negativo del tempo morto può esser corretto compensando la risposta via software. È possibile fare ciò solo se è nota la larghezza d'impulso del segnale.[non chiaro] Viste le sue ridotte dimensioni, può essere usato anche per dosimetria personale.

La dinamica di questi rivelatori è abbastanza ridotta, a causa del tempo morto durante il quale avviene un conteggio, dell'ordine dei microsecondi.



Schema di un contatore Geiger

OLTRE L'ALGEBRICO! *Dal* [DAVE HESLOP]:

Often a Geiger counter will be used to measure the radioactivity of rocks. One problem with simple Geiger counters is that at high activity (high count rates) they will underestimate the true activity. This is because after a γ-ray enters the detector the system is "dead" for a short period of time during which it cannot measure. If another γ-ray enters the system during this dead-time it will not be counted and thus the observed count rate will be less than the true count rate. This situation is made worse because not only are the γ-rays which enter during the dead-time not counted, they do extend the dead-time.

Figure 3.2: *How can we find the true count rate of a Geiger counter which suffers from dead-time.*

The relationship between the observed count rate (N_{obs}) and the true count rate (N_{true}) is an exponential law equation:

OLTRE L'ALGEBRICO! *Dal* [DAVE HESLOP]:

Figure 3.2: *How can we find the true count rate of a Geiger counter which suffers from dead-time.*

The relationship between the observed count rate (N_{obs}) and the true count rate (N_{true}) is an exponential law equation:

$$N_{obs} = N_{true} e^{-N_{true}\tau}$$

where τ is the dead-time per pulse (20 x 10^{-6} seconds in old instruments). Note this is a transcendental equation which means that it cannot be rewritten in the form $N_{true} = ...$ In this exercise you will write an M-function to determine the value of N_{true} for a given input value of N_{obs} and τ.

Come si esprime in MATLAB l'esponenziale e^x ?

Command Window

```
>> doc exp % la celebre identità di Eulero
>> exp(i*pi)

ans =

 -1.000000000000000 + 0.000000000000000i

>> exp(i*pi)+1

ans =

      0.000000000000000e+00 + 1.224646799147353e-16i

>>
```

Description

`Y = exp(X)` returns the exponential e^x for each element in array X. For complex elements $z = x + iy$,

$$e^z = e^x(\cos y + i \sin y) .$$

Use `expm` to compute a matrix exponential.

Examples

Numeric Representation of e

Calculate the exponential of 1, which is Euler's number, *e*.

```
exp(1)
```

```
ans = 2.7183
```

Euler's Identity

Euler's identity is the equality $e^{i\pi} + 1 = 0$.

Compute the value of $e^{i\pi}$.

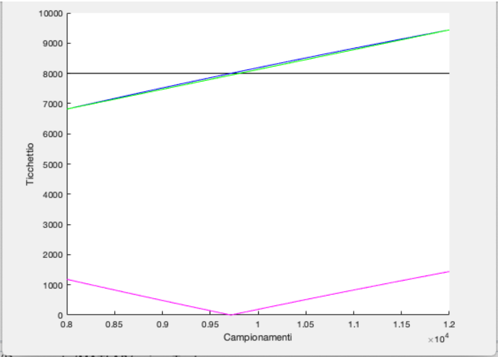
```
Y = exp(1i*pi)
```

```
Y = -1.0000 + 0.0000i
```

```
Ntrue =

        9716

>>
```



Editor – /Users/eugenioomodeo/Documents/MATLAB/geigerTest.m

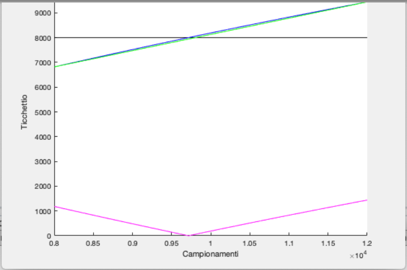
geiger.m | geigerTest.m | digitCompare.m | crackSafe.m | crackWeakSafe.m | classifyingSediments.m | nsceltek.m | daOttimizzare

```
clc,clear,close all,
Nobs=8000;        % numero di click al secondo osservati
tau=20e-6;        % tempo morto per impulso, espresso in secondi
Ntest=[8e3:12e3];  % da 8000 a 12000
Ntrue=geiger(Nobs,tau,Ntest) % la stima del valore reale più calzante
```

```
Ntrue =

        9716

>>
```

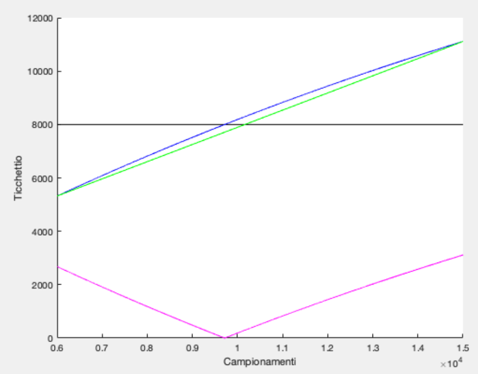


Editor – /Users/eugenioomod

geiger.m | geigerTest. | ents.m | nsceltek.m | daOttimizzare.m | bi

```matlab
function Ntrue = geiger(Nobs,tau,Ntest) % Da [Heslop2012], pagg.146--148
% `Nobs` esprime il conto osservato dell'attività
% (emissione di raggi gamma);
% `tau` esprime il tempo morto per ciascun impulso
% (circa 20e-6 secondi con strumenti vecchi);
% `Ntest` esprime un array di valori da cui
% va scelta la predizione che è più vicina al
% conto osservato `Nobs`.
Ncalc = Ntest .* exp( -Ntest * tau ); % `cps` predetti
Ndiff = abs(Ncalc-Nobs); % differenza rispetto a quanto osservato
figure, hold on, plot([Ntest(1),Ntest(length(Ntest))],[Nobs,Nobs],'k');
plot(Ntest,Ncalc,'b'), xlabel('Campionamenti'), ylabel('Ticchettio'),
plot([Ntest(1),Ntest(length(Ntest))],[Ncalc(1),Ncalc(length(Ncalc))],'g'),
plot(Ntest,Ndiff,'m'),
[Ndiff_min,idx] = sort(Ndiff); % localizzare la minima distanza
Ntrue=Ntest(idx(1)); % scelta del valore che calza meglio di tutti
```

Visualizzazione di quanto precede

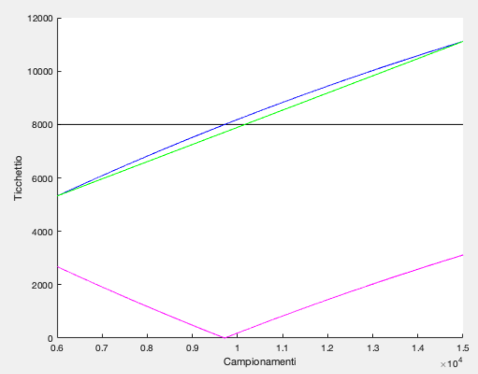


k: rilevazione effettiva del numero dei '*click*';

b: rilevazioni che corrisponderebbero a diversi valori di irradiamento;

g: retta che collega primo e ultimo valore che verrebbero rilevati;

m: scarto fra i valori che verrebbero rilevati in corrispondenza a diversi irradiamenti e quanto effettivam. rilevato.



k: rilevazione effettiva del numero dei '*click*';

b: rilevazioni che corrisponderebbero a diversi valori di irradiamento;

g: retta che collega primo e ultimo valore che verrebbero rilevati;

m: scarto fra i valori che verrebbero rilevati in corrispondenza a diversi irradiamenti e quanto effettivam. rilevato.