Soluzione di sistemi lineari

- Metodi per la soluzione di sistemi di equazioni lineari:
  - Eliminazione di variabili
  - Metodo di Cramer
  - Matrice inversa
- Tipi di sistemi:
  - Sistemi determinati
  - Sistemi indeterminati
  - Sistemi sovradeterminati

Esistenza delle soluzioni

- Quante soluzioni?
  - 1 se singolare
  - 0 o infinite se non singolare

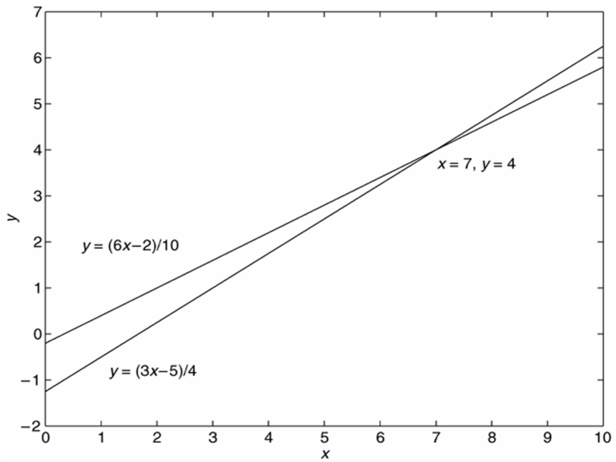
$$\begin{cases} 6\ x\ -10\ y = 2 \\ 3\ x\ \ -4\ y = 2 \end{cases}$$

Abbiamo una soluzione: intercetta in y = 4, x = 7

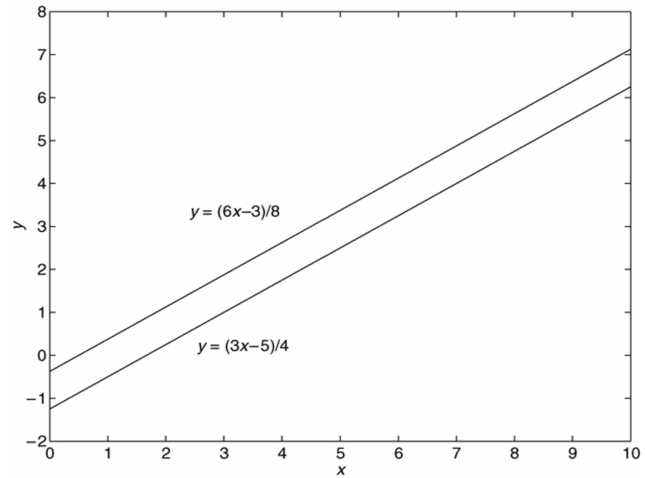
$$\begin{cases} 3\ x\ \ -4\ y = 5 \\ 6\ x\ \ -8\ y = 3 \end{cases}$$

Nessuna soluzione: Rette parallele!

Primo sistema:



Secondo sistema:



Rappresentazione come matrici

- Sistema di *n* equazioni in *m* incognite (le *x*)

$$
\begin{array}{llcll}
a_{11}x_1 & +a_{12}x_2 & \dots & +a_{1m}x_m & = & b_1 \\
a_{21}x_1 & +a_{22}x_2 & \dots & +a_{11}x_m & = & b_2 \\
\vdots & & \ddots & & & \vdots \\
a_{n1}x_1 & +a_{n2}x_1 & & +a_{nm}x_m & = & b_n
\end{array}
$$

- Per esempio:

$$
\begin{array}{llll}
2x_1 & +3x_2 & +3x_3 & =7 \\
4x_1 & +2x_2 & +9x_3 & =5 \\
6x_1 & -7x_2 & +2x_3 & =1
\end{array}
$$

- In forma matriciale: $\mathrm{A} * \mathrm{x} = \mathrm{b}$
  con

$$
\mathrm{A} = \begin{bmatrix} 2 & 3 & 3 \\ 4 & 2 & 9 \\ 6 & -7 & 2 \end{bmatrix} \quad x = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} \quad b = \begin{bmatrix} 7 \\ 5 \\ 1 \end{bmatrix}
$$

$$
\begin{array}{llcll}
a_{11}x_1 & +a_{12}x_2 & \dots & +a_{1m}x_m & = & b_1 \\
a_{21}x_1 & +a_{22}x_2 & \dots & +a_{11}x_m & = & b_2 \\
\vdots & & \ddots & & & \vdots \\
a_{n1}x_1 & +a_{n2}x_1 & & +a_{nm}x_m & = & b_n
\end{array}
$$

- In generale: $\mathrm{A} * \mathrm{x} = \mathrm{b}$

  Vettori colonna (minuscola)

  - A è una n per m
  - x è una m per 1
  - b è una n per 1

Esistenza delle soluzioni

Nel sistema di n equazioni in n incognite (A quadrata)

A x = b

- Se il determinante di A è nullo non c'è una soluzione unica!
- Possono esserci 0 o infinite soluzioni, a seconda del valore di b

Metodi di soluzione

- Metodo di Gauss
  - Sommiamo un'equazione (moltiplicata per un coefficiente) ad un'altra
  - Eliminiamo una variabile alla volta.
- Matrice Inversa
- Determinante di Cramer

Metodo di Gauss in Matlab

- Si usa l'operatore divisione sinistra: x = A\b. Per Esempio,

```
>> A = [6, -10; 3, -4]; b = [2; 5];
>> x = A\b
x =
  7  4
```


- Questo metodo funziona sicuramente se abbiamo n equazioni ed n incognite (A quadrata) e det(A) diverso da 0.
- Se det(A) = 0 o se A non è quadrata dobbiamo usare altri metodi (anche se in certi casi la divisione sinistra funziona uguamente).

Metodo di Cramer

- L' i-esima soluzione del sistema A x = b sono ricavabili con l'espressione:

$$x_i = \frac{\det(A_i)}{\det(A)}$$

- Dove Ai è la matrice ottenuta dalla matrice A sostituendo la i-esima colonna con il vettore b
- Non è comodo per i computer ma ci dice perché det(A) deve essere non nullo!

Metodo della matrice inversa

- Il comando inv(A) calcola l'inversa di A
- Per risolvere $\begin{cases} 2x + 9y = 5 \\ 3x - 4y = 7 \end{cases}$

```
>>A = [2,9;3,-4];
>>b = [5;7]
>>x = inv(A)*b
x =
   2.3714
   0.0286
```

Se A è singolare inv(A) ci restituisce un messaggio di errore!!!

Sistemi indeterminati

- In un sistema indeterminato ci sono meno equazioni che incognite (in generale)
- In pratica non abbiamo abbastanza informazioni per risolverlo
- Possono esistere infinite soluzioni, in cui una o più incognite dipendono dalle altre
- Con questi sistemi Cramer e la matrice inversa non funzionano.

Un esempio banale

Abbiamo il sistema: x +3 y = 6

- Possiamo trovare x in funzione di y:
  x = 6 – 3 y , ma niente di più
- Abbiamo 2 incognite ed una sola equazione.
- Il metodo della divisione a sinistra troverà una soluzione con una variabile nulla:

```
>>A = [1, 3]; b = 6;
>>solution = A\b
solution =
  0
  2
```

Cioè x=0, y=2

- Il sistema può essere indeterminato anche con A, quadrata, se det(A)=0.
- Anche in questo caso potremmo avere infinite soluzioni (oppure nessuna)
- In questo caso Cramer non funziona, e la divisione sinistra ci da un messaggio di errore (A è singolare)
- Possiamo usare la
  pseudo-inversa: $A^+ = (A^T A)^{-1} A^T$
- In Matlab:
  `x = pinv(A) * b`
  Ci fornisce la soluzione di norma minima.

La funzione rref

- Nei sistemi indeterminati possiamo dare una soluzione esprimendo alcune incognite in funzione delle altre.
- Possiamo sempre ridurre un sistema in questa forma (reduced row echelon form, matrice a righe ridotte)
- Il comando rref([A b]) fornisce in uscita la matrice [C d] che identifica il sistema ridotto C x = d

Sistemi sovradeterminati

- un sistema si dice sovradeterminato quando ci sono più equazioni che incognite (in generale)
- In questi casi Cramer e l'inversa non funzionano perché A non è quadrata
- In alcuni casi può esistere una soluzione, ottenibile con la divisione sinistra x = A \ b

- In altri casi non esiste una soluzione esatta
- Allora la divisione a sinistra fornisce la soluzione nel senso dei "minimi quadrati"
- Però Matlab non ci dice se è esatta o dei minimi quadrati.
- Per scoprirlo dobbiamo verificare il rango di A e il rango di [A b], se sono uguali la soluzione è esatta, altrimenti è stata trovata la soluzione ai minimi quadrati.

Un esempio:

- Dobbiamo trovare l'equazione della retta

  y = m x + b

  che passi per i seguenti punti:

  (0,2) (5,6) (10,11)
- Le incognite sono i coefficienti (m, b) e il sistema è:

```
 0 m + b = 2
 5 m + b = 6
10 m + b = 11
```

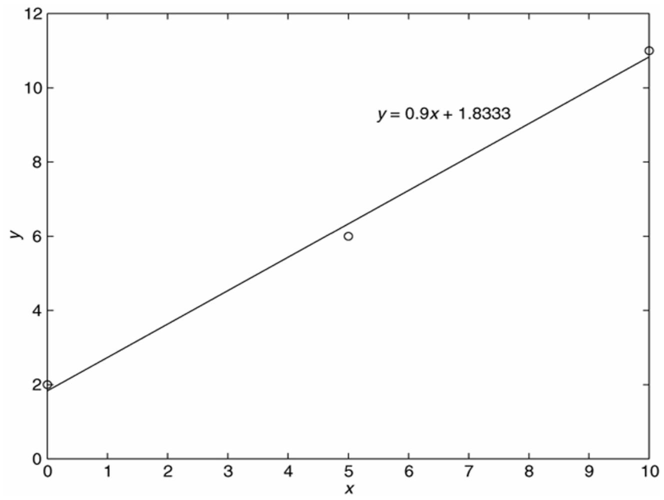
Le matrici sono:

```
A = [0 1;5 1;10 1]
b = [ 2 ; 6 ; 11 ]
```

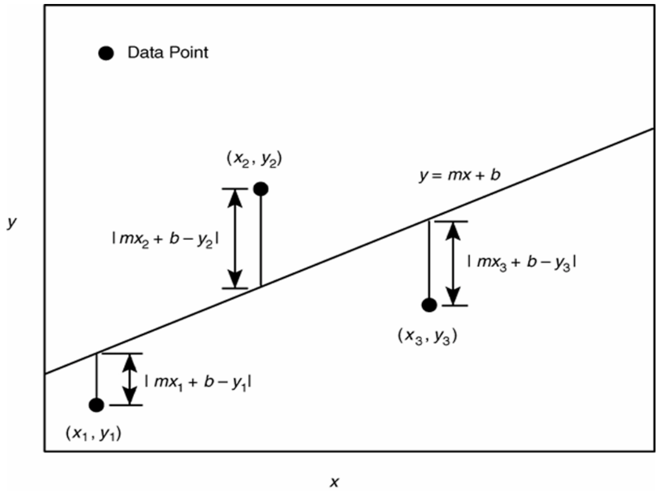
```
>> A = [0 1;5 1;10 1] ;b = [ 2 ; 6 ; 11 ];
>> rank(A)
ans =
     2
>> rank([A b])
ans =
     3
>> A\b
ans =
    0.9000
    1.8333

>> A*ans
ans =
    1.8333
    6.3333
   10.8333
```

Ecco i coefficienti:
m
b

La nostra soluzione:



Perché minimi quadrati?

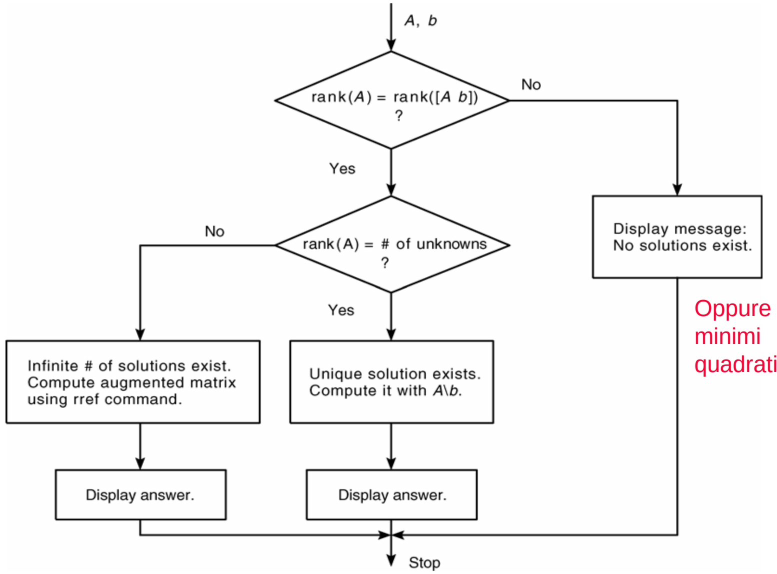


Una condizione più generale

- Possiamo dire che il sistema A x = b con m equazioni ed n incognite ha soluzione esatta se e solo se

  rank (A) == rank ([A b])

- Se questa è soddisfatta e:
  - Rank(A)=n la soluzione è unica
  - Rank(A)<n ci sono infinite soluzioni e abbiamo rank(A) incognite che dipendono dalle altre



Oppure minimi quadrati!


```
% Script file lineq.m
% Solves the set Ax = b, given A and b.
% Check the ranks of A and [A b].
if rank(A) == rank([A b])
   % The ranks are equal.
   Size_A = size(A);
   % Does the rank of A equal the number of
     %    unknowns?
   if rank(A) == size_A(2)
      % Yes. Rank of A equals the number of
      %  unknowns.
      disp('There is a unique solution,…
          which is:')
      x = A\b % Solve using left division.
```



```
 else
      % Rank of A does not equal the number
      % of unknowns.
      disp('There is an infinite number of …
          solutions.')
      disp('The augmented matrix of the …
          reduced system is:')
      rref([A b]) % Compute the augmented
                  % matrix.
   end
else
   % The ranks of A and [A b] are not equal.
   disp('There are no solutions.')
end
```