

Appunti di Analisi Numerica

Roberto Castellotti

November 27, 2020

Contents

1	Teoria degli errori	2
1.1	Tipi di errori	3
1.1.1	Errore Assoluto	3
1.1.2	Errore Relativo	3
1.1.3	Condizionamento	3
2	Numeri di macchina	4
2.1	Rappresentazione in virgola fissa (fixed point)	4
2.2	Rappresentazione in virgola mobile (floating point)	4
2.3	Errore algoritmico	5
3	Errore algoritmico	8
4	Rappresentazione degli algoritmi tramite grafi	9
5	Sistemi Lineari dal punto di vista numerico	11
5.1	Eliminazione Gaussiana	12
6	Strategia di Pivoting Parziale	14
6.1	Applicazioni	15
7	Condizionamento del problema	15
7.1	Norme vettoriali	15
8	Teorema	16
9	Norme Matriciali	17
10	Definizione norma matriciale indotta	17
11	Dimostrazione	18
12	Esempi di norme indotte	18

13 Teorema	19
14 Teorema	19
15 Errore inerente nei sistemi lineari	19
15.1 Dimostrazione	20
15.2 Matrice di Hilbert	21
15.3 Matrice di Wilkinson	21
15.4 Matrice di Lehmer	21
16 Collassamento sull' origine	31
17 Trasformazione identica (lascia tutto fermo)	32
18 s	32
19 matrice uguale meno id	32
20 Interpretazioni geometriche delle operazioni algebriche	32
20.1 Somma tra Matrici	32
20.2 Scalatura di una matrice	32
20.3 Prodotto righe per colonne	33
21 Inversa della matrice	34
22 Nucleo e Immagine	34
23 Nuclei e Immagini di matrici viste	36
24 Quindi	37
25 Isometrie	38
26 Esempi notevoli di isometrie	40
27 Rotazioni di Givens	42
28 Proprieta' della fattorizzazione QR	49
29 Riflessioni di Householder	49
30 Fattorizzazione QR	53

1 Teoria degli errori

Ogni volta che si tratta un **problema reale** per prima cosa si deve tradurre quanto osservato in un **modello matematico**, che nella maggior parte dei casi e' di **tipo continuo**, i.e: $\int_a^b x dx$, risulta evidente che sia impossibile darlo in

input ad un calcolatore, e' quindi necessaria una **discretizzazione** del modello, vale a dire una conversione in un **numero finito di dati** che genera un **numero finito di risultati**, solo in questo momento e' possibile applicare un algoritmo di risoluzione del problema.

1.1 Tipi di errori

Ogni trasformazione del problema comporta una introduzione di **errore**, che puo' anche essere introdotto in fase di misurazione preliminare, si distinguono 3 tipi di errori:

- **errori analitici:** nascono nel momento di **modellizzazione / discretizzazione**
- **errori inerenti:** errori di misurazione dei dati (sono SEMPRE presenti)
- **errori algoritmici:** sono dovuti alle approssimazioni nello sviluppo di algoritmi di risoluzione

Sommando i diversi errori si ottiene l' errore finale.

1.1.1 Errore Assoluto

L' errore assoluto e' definito come $\delta = \tilde{x} - x$, dove \tilde{x} rappresenta la **versione del risultato perturbata** dall' errore, mentre x rappresenta il risultato esatto.

NB. e' possibile esprimere l' errore attraverso con valore assoluto, ignorando in questo modo eccesso/difetto dell' errore.

1.1.2 Errore Relativo

Nell' esprimere un errore e' importante considerare su cosa stiamo misurando l' errore, nel calcolo dell' errore si introduce quindi l' ordine di grandezza del valore esatto.

L' **errore relativo** e' definito come $\epsilon = \frac{\tilde{x}-x}{x}$.

1.1.3 Condizionamento

Se consideriamo dati in input $(x_1 \dots x_n)$ e risultati $(y_1 \dots y_m)$ si ha come errore in input: $r_j = \frac{\tilde{x}_j - x_j}{x_j}$ e come errore in output: $\epsilon_i = \frac{\tilde{y}_i - y_i}{y_i}$, si definisce **condizionamento** $c = \frac{r_j}{\epsilon_i}$

$$\begin{cases} x + y = 2 \\ 1001x + 1000y = 2001 \end{cases} = \begin{cases} x = 1 \\ y = 1 \end{cases}$$

$$\begin{cases} (1 + \frac{1}{x})x + y = 2 \\ 1001x + 1000y = 2001 \end{cases} = \begin{cases} \tilde{x} = -\frac{1}{9} \\ \tilde{y} = \frac{1901}{900} \end{cases}$$

Condizionamento per la X :

$$\frac{\frac{(-\frac{1}{9} - 1)}{\frac{1}{100}}}{\frac{1}{1} - 1} = -\frac{1000}{9} \approx 100$$

Un condizionamento di ≈ 100 significa che amplifichiamo ogni errore di 100 volte, questo e' molto grave.

Consideriamo una funzione che ha input x e output $f(x)$, perturbando l' input otteniamo come input \tilde{x} e come output $\tilde{y} = f(\tilde{x})$, il condizionamento e' quindi:

$$c = \frac{\frac{(\tilde{y} - y)}{y}}{\frac{\tilde{x} - x}{x}} = \frac{f(\tilde{x}) - f(x)}{f(x)} \cdot \frac{x}{\tilde{x} - x}$$

La parte evidenziata rappresenta il **rapporto incrementale**, questo ci suggerisce di considerare:

$$\lim_{\tilde{x} \rightarrow x} c = \frac{x f'(x)}{f(x)}$$

questo valore, che e' una buona stima per perturbazioni piccole e' chiamato **coefficiente di amplificazione**

foglio 1 quaderno 1

2 Numeri di macchina

2.1 Rappresentazione in virgola fissa (fixed point)

La rappresentazione **fixed point** prevede la seguente sintassi:

$$d_m B^m + \dots + d_1 B^1 + d_0 B^0 + d_{-1} B^{-1} + \dots + d_{-s} B^{-s}$$

2.2 Rappresentazione in virgola mobile (floating point)

La rappresentazione **floating point** prevede la seguente sintassi:

$$0.d_1 d_2 d_3 \dots d_r \cdot B^p$$

e prevede che il primo numero dopo la virgola sia diverso da zero (per garantire univocita' rappresentativa)

$$\mathbb{F} = (B, t, m, M) = \left\{ x \in \mathbb{R} : x = \pm f B^p, -m \leq p \leq M, f = \sum_{i=1}^t d_i B^{-i} \right\}$$

Valori di precisione di macchina:

- Singola precisione del C: $\mathcal{F}(2, 23, 128, 127)$
- Doppia precisione di Matlab: $\mathcal{F}(2, 52, 1024, 1023)$

Per convertire un numero a un numero di macchina esistono due strategie:

- **chopping**: ignora le cifre che escono dallo spazio di rappresentazione
- **rounding**: controlla se la prima cifra fuori da t e' maggiore o minore della meta' della base, se e' maggiore viene aggiunta una unita' nell' ultima cifra disponibile

2.3 Errore algoritmico

Ogni volta che memorizziamo un numero in macchina (ogni passaggio algoritmico quindi) commettiamo errore che continua a sommarsi.

Rappresentando un numero $x = \pm f B^p$ in macchina otteniamo $\tilde{x} = \pm \tilde{f} B^p$, il **massimo errore** (scarto tra le mantisse del valore effettivo e del valore rappresentato in macchina) e':

$$|\tilde{f} - f| \leq \frac{1}{2} B^{-t}$$

mentre l' **errore relativo** e':

$$\frac{|\tilde{x} - x|}{|x|} = \pm \frac{\tilde{f} B^p - (\pm f B^p)}{|\pm f B^p|} = \frac{B^p |\tilde{f} - f|}{B^p f}$$

Sappiamo adesso maggiorare questo errore imponendo la condizione espressa precedentemente:

$$\frac{|\tilde{f} - f|}{f} \leq \frac{1}{2} B^{-t}$$

Osserviamo che la minima mantissa f che possiamo scrivere e' B^{-1} (il primo valore dopo la virgola non puo' essere zero), il massimo errore relativo nel rappresentare un numero (**precisione di macchina (u)**) e':

$$\frac{\frac{1}{2} B^{-t}}{B^{-1}} = \frac{1}{2} B^{-t+1}$$

I valori al di sotto della precisione di macchina sono **elementi neutri**

Se usiamo una funzione del tipo $\sqrt{\cdot}$, \log , \sin sappiamo che l' errore inerente e' $e_{out} \simeq c \cdot e_{in}$

Errore algoritmico della somma

$a, b \rightarrow a + b$ in realta' corrisponde a $\tilde{a}, \tilde{b} \rightarrow \tilde{a} + \tilde{b}$, questo valore verra' poi arrotondato, introducendo **errore relativo**, il valore ottenuto nella macchina e' $(\tilde{a} + \tilde{b})(1 + \varepsilon)$

Misuriamo l' errore considerando $\tilde{a} = a(1 + \varepsilon_a)$ e $\tilde{b} = b(1 + \varepsilon_b)$:

$$\begin{aligned} \varepsilon_{a+b} &= \frac{[a(1 + \varepsilon_a) + b(1 + \varepsilon_b)](1 + \varepsilon) - (a + b)}{a + b} = \\ &= \frac{[a + a\varepsilon_a + b + b\varepsilon_b](1 + \varepsilon) - a - b}{a + b} = \\ &= \frac{\cancel{a} + a\varepsilon_a + \cancel{b} + b\varepsilon_b + a\varepsilon + a\varepsilon_a\varepsilon + b\varepsilon + b\varepsilon_b\varepsilon - \cancel{a} - \cancel{b}}{a + b} = \end{aligned}$$

Le quantita' evidenziate sono molto piccole, portiamo avanti i conti al primo ordine(\doteq), ovvero trascuriamo tutti i termini paragonabili a $\simeq u^2$

$$\varepsilon_{a+b} \doteq \varepsilon_a \left(\frac{a}{a+b} \right) + \varepsilon_b \left(\frac{b}{a+b} \right) + \varepsilon$$

I valori evidenziati sono **coefficienti di amplificazione**
Si ottengono risultati del tutto analoghi per la sottrazione:

$$\varepsilon_{a-b} \doteq \varepsilon_a \left(\frac{a}{a-b} \right) - \varepsilon_b \left(\frac{b}{a-b} \right) + \varepsilon$$

Quanto puo' essere grande $\frac{a}{a+b}$?

- $a, b > 0 \rightarrow \frac{a}{a+b} < 1$ (**riduzione dell' errore**) (vale anche se sono entrambi negativi)
- caso estremo **cancellazione**: $a \simeq -b \rightarrow a + b \ll a$ (numeratore \ll denominatore) (**amplificazione dell' errore**)

Esempio 1. $B = 10, t = 6a = 0.123456b = -0.123454a + b = 0.000002 \cdot 10^0$, questa rappresentazione non e' valida in **floating point**, scriviamo quindi $0.200000 \cdot 10^{-5}$, ma questi zeri provengono da valori che prima non conosceamo, sono arbitrarie e rischiamo di avere un errore anche molto significativo dalla seconda cifra in poi, il coefficiente di amplificazione e' $\frac{a}{a-b} \simeq \frac{0.1}{2 \cdot 10^{-6}} = 50000$

Errore algoritmico della moltiplicazione

$$\tilde{a} = a(1 + \varepsilon_a), \tilde{b} = b(1 + \varepsilon_b)$$

$$\tilde{a}\tilde{b}(1 + \varepsilon) = a(1 + \varepsilon_a) \cdot b(1 + \varepsilon_b)(1 + \varepsilon) \doteq \quad (1)$$

$$\doteq ab(1 + \varepsilon_a + \varepsilon_b + \varepsilon_a\varepsilon_b)(1 + \varepsilon) \doteq \quad (2)$$

$$\doteq ab(1 + \varepsilon_a + \varepsilon_b + \varepsilon + \dots) \doteq \quad (3)$$

$$\varepsilon_{a \cdot b} = \frac{\cancel{ab}(1 + \varepsilon_a + \varepsilon_b + \varepsilon) - \cancel{ab}}{\cancel{ab}} = \varepsilon_a + \varepsilon_b + \varepsilon \quad (4)$$

La propagazione dell' errore e' molto meno influente rispetto alla somma.

Errore algoritmico della divisione

$$\tilde{a} = a(1 + \varepsilon_a), \tilde{b} = b(1 + \varepsilon_b)$$

$$\begin{aligned} \frac{\tilde{a}}{\tilde{b}}(1 + \varepsilon) &= \frac{a(1 + \varepsilon_a)(1 + \varepsilon)}{b(1 + \varepsilon_b)} \doteq \\ &\doteq \frac{a(1 + \varepsilon_a + \varepsilon)}{b(1 + \varepsilon_b)} \cdot \frac{1 - \varepsilon_b}{1 - \varepsilon_b} \doteq \\ &\doteq \frac{a(1 + \varepsilon_a + \varepsilon)(1 - \varepsilon_b)}{b(1 - \cancel{\varepsilon_b})} \doteq \\ &\doteq \frac{a(1 + \varepsilon_a + \varepsilon - \varepsilon_b)}{b} \end{aligned}$$

quindi

$$\begin{aligned} \varepsilon_{a/b} &= \frac{\frac{a(1 + \varepsilon_a + \varepsilon - \varepsilon_b)}{b} - \frac{a}{b}}{a/b} \doteq \\ \varepsilon_{a/b} &= 1 + \varepsilon_a + \varepsilon - \varepsilon_b - 1 \\ \varepsilon_{a/b} &= \varepsilon_a - \varepsilon_b + \varepsilon \end{aligned}$$

La propagazione dell' errore e' simile a quella per la moltiplicazione.

3 Errore algoritmico

Consideriamo un esercizio svolto in precedenza ($f(x) = x^2 - 7x$), sappiamo che l' errore inerente e' $\varepsilon_{in} = \frac{2x-7}{x-7}\varepsilon_x$.

Consideriamo adesso 2 algoritmi per calcolare la funzione (prevediamo 2 errori diversi):

- **alg1:** $x \rightarrow q := x \cdot x, p := 7x \rightarrow y_1 = q - p$
- **alg2:** $x \rightarrow d := x - 7 \rightarrow y_2 = xd$
- **alg1:**

$q = x \cdot x$ errore:

$$\varepsilon_q^{(tot)} = \varepsilon_x^{(tot)} + \varepsilon_q^{(tot)} + \varepsilon_q = 2\varepsilon_x + \varepsilon_q$$

ε_q rappresenta l' errore locale, ε_x e' gia' stato misurato nell' errore inerente

$p = 7 \cdot x$ errore:

$$\varepsilon_p^{(tot)} = \underbrace{\varepsilon_7^{(tot)}}_0 + \varepsilon_x^{(tot)} + \varepsilon_p = \varepsilon_x + \varepsilon_p$$

$y_1 = q - p$ errore:

$$\begin{aligned} & \frac{q}{q-p} \varepsilon_q^{(tot)} - \frac{p}{q-p} \varepsilon_p^{(tot)} + \varepsilon_{y_1} = \\ & = \frac{q}{q-p} \varepsilon_q - \frac{p}{q-p} \varepsilon_p + \varepsilon_{y_1} = \\ & = \frac{x^2}{x^2-7x} \varepsilon_q - \frac{7x}{x^2-7x} \varepsilon_p + \varepsilon_{y_1} = \\ & = \frac{x}{x-7} \varepsilon_q - \frac{7}{x-7} \varepsilon_p + \varepsilon_{y_1} = \end{aligned}$$

Osserviamo che per $x \approx 7$ ε_{alg1} tende a esplodere

- **alg2**

$d = x - 7$ errore:

$$\varepsilon_d^{(tot)} = \frac{x}{x-7} \varepsilon_x^{(tot)} - \underbrace{\frac{7}{x-7} \varepsilon_7^{(tot)}}_0 + \varepsilon_d = \varepsilon_d$$

$y_2 = x \cdot d$ errore:

$$\varepsilon_x^{(tot)} + \varepsilon_d^{(tot)} + \varepsilon_{y_2} = \varepsilon_d + \varepsilon_{y_2}$$

Nel caso peggiore l' errore e' 2 volte la precisione di macchina.

$$|\varepsilon + alg2| \ll |\varepsilon_{alg1}| \approx |\varepsilon_{in}|$$

Questo ci permette di dire che **alg2** e' piu' stabile di **alg1**, del resto **alg1** forzava una cancellazione per $x \approx 7$ (lo sapevamo gia' perche' avevamo detto che il problema era malcondizionato), **alg2** forza una cancellazione, ma siccome questa amplifica tutti gli errori commessi precedentemente e prima non avevamo svolto operazioni non abbiamo una notevole amplificazione di errore. Questo ci insegna a evitare le cancellazioni, se non si puo' dobbiamo farle appena possibile.

4 Rappresentazione degli algoritmi tramite grafi

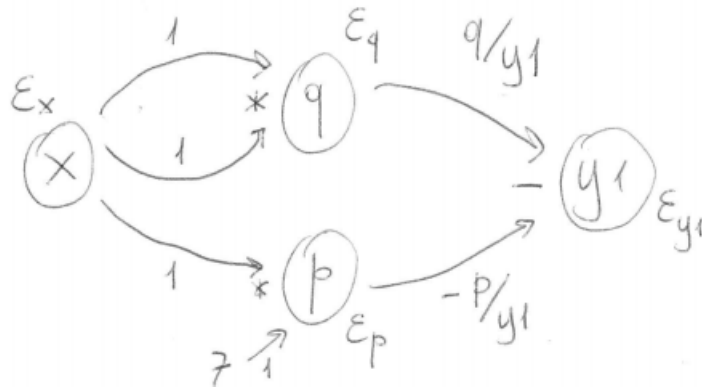
Un modo molto efficiente per rappresentare gli algoritmi e' quello dei grafi formati secondo le seguenti regole:

- i nodi rappresentano le variabili
- gli archi rappresentano le operazioni che coinvolgono i nodi
- i nodi sono etichettati con le operazioni locali

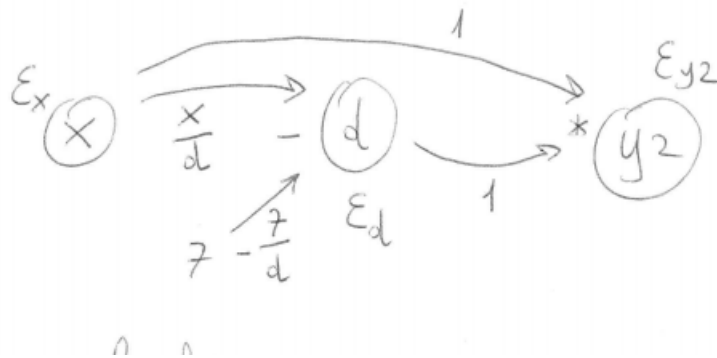
- gli archi sono etichettati con i coefficienti di amplificazione (di operazioni o funzioni)

Consideriamo ancora una volta $f(x) = \underbrace{x^2}_q - \underbrace{7x}_p$:

- alg1



- alg2



Per calcolare l' errore seguiamo la seguente formula:

$$\varepsilon_{loc} = \left(\sum_{\text{cammini}} \prod_{\text{archi seguiti}} \cdot \text{etichette} \right)$$

Per esempio, per l' algoritmo 1 abbiamo:

$$\begin{aligned} \varepsilon_{alg1} &= \varepsilon_q \left[\frac{q}{y_1} \right] + \varepsilon_p \left[-\frac{p}{y_1} \right] + \varepsilon_{y1}[1] = \\ &= \varepsilon_{alg1} = \varepsilon_q \left[\frac{x^2}{x^2 - 7x} \right] + \varepsilon_p \left[-\frac{7x}{x^2 - 7x} \right] + \varepsilon_{y1}[1] = \end{aligned}$$

$$= \varepsilon_{alg1} = \varepsilon_q \left[\frac{x}{x-7} \right] + \varepsilon_p \left[-\frac{7}{x-7} \right] + \varepsilon_{y1}[1]$$

Mentre per l' algoritmo 2:

$$\varepsilon_{alg2} = \varepsilon_d [1] + \varepsilon_{y2} [1]$$

Nel caso in cui esistano piu' cammini li sommiamo all' interno delle parentesi quadre.

5 Sistemi Lineari dal punto di vista numerico

Per indicare una matrice generica usiamo $A = \begin{bmatrix} a_{11} & \dots & a_{1n} \\ \vdots & \ddots & \\ a_{m1} & & a_{nn} \end{bmatrix} \in \mathbb{R}^{m \times n}$ Con-

sideriamo sistemi lineari quadrati, quindi $A \in \mathbb{R}^{m \times n}$ e il sistema lineare $Ax = b$ con $x, b \in \mathbb{R}^n$ e una matrice A invertibile in modo che il sistema abbia una e una sola soluzione, ovvero $\det(A) \neq 0$.

Data in input la matrice dei coefficienti del sistema vogliamo calcolare la soluzione.

Il procedimento standard per risolvere un sistema generico e' il seguente:

$$Ax = B \xrightarrow{\text{Gauss}} \tilde{A}x = \tilde{b} \text{ triangolare superiore}$$

Considerando la matrice triangolare

$$\tilde{A} = \begin{bmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ 0 & a_{22} & & \vdots \\ \vdots & 0 & \ddots & \vdots \\ 0 & \dots & 0 & a_{nn} \end{bmatrix}, x = \begin{bmatrix} x_1 \\ \vdots \\ x_n \end{bmatrix} \text{ e } b = \begin{bmatrix} b_1 \\ \vdots \\ b_n \end{bmatrix}$$

Possiamo scrivere il sistema come:

$$\begin{cases} a_{11}x_1 + \dots + a_{1n}x_n = b_1 \\ a_{22}x_2 + \dots + a_{2n}x_n = b_2 \\ a_{n-1,n-1}x_{n-1} + a_{n-1,n}x_n = b_{n-1} \\ a_{nn}x_n = b_n \Rightarrow x_n = \frac{b_n}{a_{nn}} \end{cases}$$

Ora applichiamo il **metodo di sostituzione all' indietro**

$$\begin{aligned} x_n &= b_n/a_{nn} \\ \text{for } i &= n-1, n-2, \dots, 2, 1 : (\text{considera } i\text{-esima equazione}) \\ x_i &:= \frac{b_i - \sum_{j=i+1}^n a_{ij} \cdot x_j}{a_{ii}} \end{aligned}$$

Dobbiamo assicurarci che i denominatori siano $\neq 0$, questo ci e' garantito dal fatto che in una matrice triangolare superiore $\det(A) = a_{11}a_{22}a_{nn} \neq 0$, la I-esima equazione: $a_{11}x_1 + a_{1,2}x_2 + \dots + a_{1,n}x_n = b_1$, a ogni passaggio tutti i valori diversi da x_i sono termini noti (li abbiamo gia' incontrati nel ciclo). Risulta evidente che questo algoritmo abbia complessita' in $\Theta(n^2)$.

5.1 Eliminazione Gaussiana

Primo passo per la risoluzione di un sistema lineare.

$$A^{(1)} = \begin{bmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & a_{22} & \dots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & \dots & a_{nn} \end{bmatrix} \longrightarrow A^{(n)} = \begin{bmatrix} a_{11}^{(n)} & a_{12}^{(n)} & \dots & a_{1n}^{(n)} \\ 0 & a_{22}^{(n)} & \dots & a_{2n}^{(n)} \\ 0 & 0 & \ddots & \vdots \\ 0 & 0 & \dots & a_{nn}^{(n)} \end{bmatrix}$$

Un passaggio generico k e' rappresentato da una matrice in cui:

- sono state annullate $k - 1$ colonne
- la k -esima colonna viene azzerata
- $a_{kk}^{(k)}$ e' detto **pivot**

$$A^{(k)} = \begin{bmatrix} a_{11}^{(k)} & a_{12}^{(k)} & \dots & a_{1n}^{(k)} \\ 0 & a_{k-1,k-1}^{(k)} & \dots & a_{2n}^{(k)} \\ 0 & 0 & a_{kk}^{(k)} & \vdots \\ 0 & 0 & \dots & a_{nn}^{(k)} \end{bmatrix}$$

Il passaggio che facciamo e': $A^{(k)} \rightsquigarrow A^{(k+1)}$ in cui azzeriamo la k -esima colonna, assumiamo che il pivot $a_{kk}^{(k)}$ sia sempre diverso da zero, per annullare tutti i valori "sotto" dobbiamo moltiplicare per un fattore m_{ik} tutti i valori della riga del pivot e sottrarre la riga alla riga "sotto":

$$i = k + 1, \dots, n : m_{ik} := \frac{a_{ik}^{(k)}}{a_{kk}^{(k)}} \rightsquigarrow \text{riga } i\text{-esima} - m_{ik}(\text{riga } k\text{-esima})$$

Scriviamo l' algoritmo:

Algorithm 1: Algoritmo riduzione di Gauss

```
for  $k = 1, \dots, n - 1$  ( $A^{(k)} \rightarrow A^{(k+1)}$ )
  pivot =  $a_{kk}^{(k)} \neq 0$ 
  for  $i = k + 1, \dots, n$ 
     $m_{ik} = a_{ik}^{(k)} / a_{kk}^{(k)}$ 
  for  $j = k + 1, \dots, n$ 
     $a_{ij}^{(k+1)} = a_{ij}^{(k)} - m_{ik} \cdot a_{kj}^{(k)}$ 
```

Il metodo di Gauss ha un costo asintotico di $\frac{n^3}{3}$. Facciamo un esempio:

$$A^{(1)} = \begin{bmatrix} 2 & -4 & 1 \\ 6 & -14 & 8 \\ -2 & 0 & 6 \end{bmatrix} b^{(1)} = \begin{bmatrix} 1 \\ -1 \\ 1 \end{bmatrix}$$

- $k=1$
- $m_{21} = \frac{6}{2} = 3 \rightsquigarrow R_2 - 3R_1$
- $m_{31} = \frac{-2}{2} = -1 \rightsquigarrow R_3 - (-1)R_1$

$$A^{(2)} = \begin{bmatrix} 2 & -4 & 1 \\ 0 & -2 & 5 \\ 0 & -4 & 7 \end{bmatrix} b^{(2)} = \begin{bmatrix} 1 \\ -4 \\ 2 \end{bmatrix}$$

- $k=2$
- $m_{32} = \frac{-4}{-2} = 2 \rightsquigarrow R_3 - 2R_2$

$$A^{(3)} = \begin{bmatrix} 2 & -4 & 1 \\ 0 & -2 & 5 \\ 0 & 0 & -3 \end{bmatrix} b^{(3)} = \begin{bmatrix} 1 \\ -4 \\ 10 \end{bmatrix}$$

Ora dobbiamo applicare l' algoritmo di risoluzione all' indietro.

$$\begin{cases} 2x_1 - 4x_2 + x_3 = 1 \\ -2x_2 + 5x_3 = -4 \\ -3x_3 = 10 \end{cases} \xrightarrow{\text{svolgimento}} \begin{cases} x_1 = -\frac{21}{2} \\ x_2 = -\frac{19}{3} \\ x_3 = -\frac{10}{3} \end{cases}$$

L' algoritmo non termina sempre, per esempio se abbiamo un pivot =0
Facciamo un esempio:

$$A^{(1)} = \begin{bmatrix} -1 & -4 & 0 \\ 2 & 8 & 1 \\ 2 & -3 & 1 \end{bmatrix} b^{(1)} = \begin{bmatrix} 2 \\ -4 \\ 7 \end{bmatrix}$$

$$m_{21} = m_{31} = -2$$

$$A^{(2)} = \begin{bmatrix} -1 & -4 & 0 \\ 0 & 0 & 1 \\ 0 & 5 & 1 \end{bmatrix} b^{(2)} = \begin{bmatrix} 2 \\ 0 \\ 11 \end{bmatrix}$$

Ora ci troviamo a processare la seconda colonna, e come possiamo osservare il pivot e' uguale a zero, la matrice $A^{(2)}$ e' ancora invertibile (verificabile facilmente con Laplace), dobbiamo quindi apportare delle modifiche all' algoritmo affinche' sia valido, in particolare e' conveniente effettuare uno scambio delle righe.

Esempio

$$A^{(1)} \begin{bmatrix} \varepsilon & 1 \\ 1 & 0 \end{bmatrix} b^{(1)} = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$$

Un calcolatore che processa questa matrice sa che ε e' diverso da zero, non fa quindi scambi di riga, processa quindi:

$$m_{21} = \frac{1}{\varepsilon}$$

$$A^{(2)} \begin{bmatrix} \varepsilon & 1 \\ 0 & -\frac{1}{\varepsilon} \end{bmatrix} b^{(2)} = \begin{bmatrix} 1 & 1 \\ 1 & -\frac{1}{\varepsilon} \end{bmatrix} \approx \begin{bmatrix} 1 \\ -\frac{1}{\varepsilon} \end{bmatrix}$$

Adesso facciamo la sostituzione all' indietro

$$x_2 = \frac{1 - \frac{1}{\varepsilon}}{-\frac{1}{\varepsilon}}$$

$$\varepsilon x_1 + x_2 = 1 \rightsquigarrow x_1 = \frac{1 - x_2}{\varepsilon}$$

Osserviamo che questa e' una cancellazione, amplifichiamo l' errore a dismisura, nel caso in cui $|\varepsilon| < u$ addirittura $x_2 = 1 \rightarrow x_1 = 0 \rightarrow x = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$

Si puo' dimostrare che l' Eliminazione Gaussiana e' stabile se gli elementi di $A^{(k)}$ $k = 2, \dots, n-1$ non sono troppo piu' grandi degli elementi della matrice di partenza. Questo ci suggerisce di effettuare lo scambio di righe non solo quando il pivot e' zero ma anche quando e' troppo piccolo. La soluzione e' ridurre il

moltiplicatore $|m_{in}| = \frac{|a_{ik}^{(k)}|}{|a_{kk}^{(k)}|}$

6 Strategia di Pivoting Parziale

L' idea e' quella di rendere il pivot piu' grande possibile in rapporto a ogni altro elemento da azzerare $a_{ik}^{(k)}$.

Definiamo i_0 come l' indice tale che $|a_{i_0,k}| = \max_{i=k,\dots,n} |a_{i,k}^{(k)}|$, scambiamo la riga k con la riga i_0 (sia in $A^{(k)}$ che in $b^{(k)}$), procediamo poi con l' algoritmo sicuro che il moltiplicatore sia il piu' basso possibile. Questo accorgimento rende l' eliminazione di Gauss un algoritmo stabile.

6.1 Applicazioni

- $(A|I) \xrightarrow{\text{Gauss}} (I|A^{-1})$ (variante Gauss-Jordan, costo n^3), questo potrebbe farci pensare a due approcci per risolvere $Ax = b$:

$$\text{Gauss: } A^{(n)}x = b^{(n)} + \text{sostituzione all'indietro costo: } \frac{n^3}{3}$$

$$\text{Gauss-Jordan: } A^{(-1)} \rightarrow x = A^{(1)} \cdot B \quad n^3$$

Tuttavia risulta evidente che Gauss-Jordan non sia una buona scelta.

- Se dobbiamo risolvere un sistema $Ax_1 = b_1, Ax_2 = b_2, Ax_3 = b_3$ possiamo fare:

$$(A|b_1|b_2|b_3) \xrightarrow{\text{Gauss}} (A^{(n)}|b_1|b_2|b_3)$$

- Definita l'ultima matrice del metodo di Gauss $A(n) := U$ e costruita una matrice L triangolare inferiore con sulla diagonale tutti 1 e per elementi i moltiplicatori con indici corrispondenti, vale il teorema della fattorizzazione LU : se il metodo di Gauss non esegue scambi, allora $L \cdot U = A$.

7 Condizionamento del problema

Nel risolvere il problema $Ax = b$ abbiamo come input A e b e il nostro output è x , studiamone adesso il condizionamento, però se abbiamo un sistema da 1M di equazioni (più che ragionevole nel 2020) abbiamo 1M di dati nei termini noti e $1M^2$ di dati nella matrice così come 1M di risultati da calcolare, il volume di dati è difficile da processare, serve uno strumento per compattare le operazioni, introduciamo quindi il concetto di **norme vettoriali** e **norme matriciali**: le prime sono modi di misurare i vettori che prendono un vettore v e associano il simbolo $\|x\|$, le seconde associano alla matrice A il simbolo $\|A\|$.

7.1 Norme vettoriali

$\|x\|_2 = \sqrt{x_1^2, x_2^2, \dots, x_n^2}$ Le norme vettoriali devono verificare alcune proprietà:

- $\forall x \in \mathbb{R}^n \quad \|x\| \geq 0$ e $\|x\| = 0 \Leftrightarrow x = 0$
- $\|\alpha x\| = |\alpha| \|x\|$ (omogeneità della norma)
- $\|x + y\| \leq \|x\| + \|y\| \forall x, y \in \mathbb{R}^n$ (disuguaglianza triangolare)

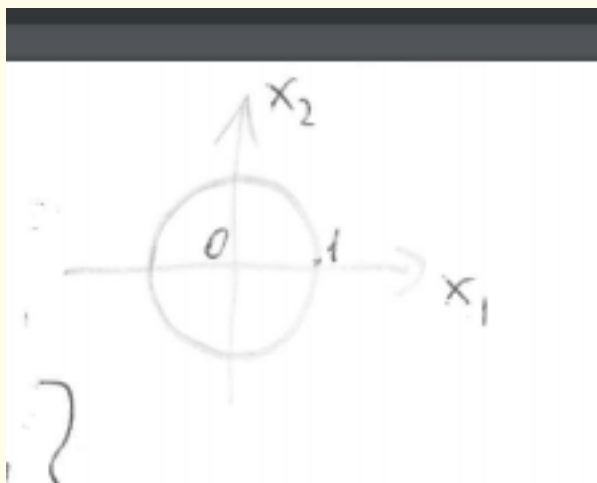
Altri esempi:

- $\|x\|_1 := |x_1| + |x_2| + \dots + |x_n|$
- $\|x\|_\infty := \max_{i=1, \dots, n} |x_i|$

Esercizio

Disegnare in \mathbb{R}^2 gli insiemi (sfere unitarie):

1. $\{x \in \mathbb{R}^2 : \|x\|_2 = 1\}$
2. $\{x \in \mathbb{R}^2 : \|x\|_1 = |x_1| + |x_2| = 1\}$
3. $\{x \in \mathbb{R}^2 : \|x\|_\infty = \max(|x_1|, |x_2|) = 1\}$



1. ?
2. $\{x \in \mathbb{R}^2 : \|x\|_1 = 1\}$ Dobbiamo distinguere in vari casi per semplificare il valore assoluto
- 3.

8 Teorema

Date due norme qualsiasi vettoriali che rispettano le proprietà precedenti diverse $\|\cdot\|^1, \|\cdot\|^2$, esistono sempre due costanti $\alpha, \beta > 0$ tali che qualsiasi coppia di vettori scelta nello spazio è confrontabile, ovvero:

$$\exists \alpha, \beta > 0 \mid \forall x \in \mathbb{R}^n \alpha \|x\|^2 \leq \|x\|^1 \leq \beta \|x\|^2$$

Per esempio consideriamo le sfere unitarie precedenti

1. $x = \begin{pmatrix} \sqrt{2}/2 \\ \sqrt{2}/2 \end{pmatrix} \rightarrow \|x\|_2 = 1, \|x\|_1 = \sqrt{2}$
2. $x = \begin{pmatrix} 1 \\ 1 \end{pmatrix} \rightarrow \|x\|_2 = \sqrt{2}, \|x\|_1 = 2, \|x\|_\infty = 2$

$\ \cdot\ ^1$	$\ \cdot\ ^2$	α	β
1	∞	1	n
1	2	1	\sqrt{n}
2	∞	1	\sqrt{n}

Table 1: Norme prese in considerazione e costanti per confrontarle

Possiamo costruire una tabella per confrontare le varie norme:
La tabella si legge (esempio ultima riga) in questo modo:

$$\|x\|_\infty \leq \|x\|_2 \leq \sqrt{n}\|x\|_\infty \quad \forall x \in \mathbb{R}^n$$

9 Norme Matriciali

Anche in questo caso vogliamo associare a una matrice una norma per compattare valori potenzialmente molto grandi, valgono le seguenti proprietà:

- $\|A\| \geq 0$ e $\|A\| = 0 \Leftrightarrow A = 0$
- $\|\alpha A\| = |\alpha|\|A\| \quad \forall \alpha \in \mathbb{R}$ (omogeneità della norma)
- $\|A + B\| \leq \|A\| + \|B\| \quad \forall A, B \in \mathbb{R}^{mn}$ (disuguaglianza triangolare)
- $\|A \cdot B\| \leq \|A\| \cdot \|B\| \quad \forall A, B \in \mathbb{R}^{nn}$

Definire norme che rispettino tutte le proprietà elencate non è semplice come per le norme vettoriali, definiamo quindi solamente la (poco usata) **norma di Frobenius**:

$$\|A\|_f := \sqrt{\sum_{ij} a_{ij}^2}$$

Per capire meglio le norme matriciali conviene definire le **norme matriciali indotte**:

Possiamo trasformare un vettore in un altro vettore moltiplicandolo per una matrice

$$A \in \mathbb{R}^{n \times n} := \begin{pmatrix} x_1 \\ \vdots \\ x_n \end{pmatrix} \in \mathbb{R}^n \rightsquigarrow A \cdot x, \text{ ovvero } \begin{pmatrix} y_1 \\ \vdots \\ y_n \end{pmatrix}$$

10 Definizione norma matriciale indotta

Fissata una norma vettoriale, la norma matriciale indotta (dalla norma scelta) è il massimo delle grandezze ottenute restringendo ai vettori in input che hanno norma uguale a uno, cioè che sono sulla sfera unitaria:

$$\|A\| := \max \|A_x\|$$

Si dimostra inoltre che ogni norma matriciale indotta verifica le 4 proprietà precedenti e inoltre verifica la seguente proprietà:

$$\|A \cdot v\| \leq \|A\| \cdot \|v\| \quad \forall A, B \in \mathbb{R}^{m \times n} \quad \forall v \in \mathbb{R}^n$$

11 Dimostrazione

Per prima cosa dobbiamo generare un vettore che sia sulla sfera unitaria, quindi dato $v \in \mathbb{R}^n$ consideriamo $x := \frac{v}{\|v\|}$ (normalizzazione, x è un **versore**, ovvero $\|x\| = 1$) per dimostrarlo poniamo $\alpha := \frac{1}{\|v\|}$, quindi $x = \alpha v \rightarrow \|x\| = \|\alpha v\| \stackrel{prop.2}{=} |\alpha| \cdot \|v\| = \frac{1}{\|v\|} \cdot \|v\| = 1$. Richiamando la definizione di norma indotta osserviamo che x è un particolare vettore della sfera unitaria, quindi $\|Ax\|$ con $\|x\| = 1$ non supera il massimo (per definizione $\|A\|$), quindi $\|Ax\| \leq \|A\|$, riscriviamo come $\|A \cdot \alpha v\| = \|\alpha \cdot Av\| \stackrel{prop.2}{=} |\alpha| \cdot \|Av\| = \frac{1}{\|v\|} \cdot \|Av\| \leq \|A\|$ CVD

12 Esempi di norme indotte

Un teorema che non dimostriamo dimostra che data una matrice $A \in \mathbb{R}^{m \times n}$ la norma a infinito della matrice è il massimo delle somme dei valori assoluti degli indici riga per riga.

$$\|A\|_\infty := \max_{\|x\|_\infty=1} \|Ax\|_\infty = \max_{i=1,\dots,m} \sum_{j=1}^n |a_{ij}|$$

Ad esempio se $A = \begin{pmatrix} 1 & 2 & 3 \\ -1 & -4 & -1 \\ 0 & -5 & 2 \end{pmatrix}$ $\|A\|_\infty = 7$

Un altro teorema equivalente afferma che:

$$\|A\|_1 := \max_{\|x\|_1=1} \|Ax\|_1 = \max_{j=1,\dots,n} \sum_{i=1}^m |a_{ij}|$$

Ad esempio se $A = \begin{pmatrix} 1 & 2 & 3 \\ -1 & -4 & -1 \\ 0 & -5 & 2 \end{pmatrix}$ $\|A\|_1 = 11$

Supponiamo di avere una norma matriciale indotta, consideriamo la matrice identità I , $\|I\| = \max_{\|x\|=1} \|I \cdot x\| = \max_{\|x\|=1} \|x\| = 1$, questa proprietà deve valere su tutte le norme indotte, la **norma di Frobenius** per la matrice identità vale \sqrt{n} , questo dimostra che non può essere indotta da nulla.

13 Teorema

Date due diverse norme matriciali (anche non indotte)

14 Teorema

Date due norme qualsiasi matriciali che rispettano le proprietà precedenti diverse $\|\cdot\|^1, \|\cdot\|^2$, esistono sempre due costanti $\alpha, \beta > 0$ tali che qualsiasi coppia di vettori scelta nello spazio è confrontabile, ovvero:

$$\exists \alpha, \beta > 0 \mid \forall x \in \mathbb{R}^{n \times n} \alpha \|x\|^2 \leq \|x\|^1 \leq \beta \|x\|^2$$

Questo come prima ci suggerisce che non ha molta importanza quale norma si sceglie di calcolare.

Ora che abbiamo introdotto lo strumento delle norme possiamo tornare a studiare l

15 Errore inerente nei sistemi lineari

Ricordiamo che l' errore inerente è quello dovuto a dati errati. Consideriamo il sistema $Ax = b$ con $A \in \mathbb{R}^{n \times n}, \det(A) \neq 0, b \in \mathbb{R}^n, x \in \mathbb{R}^n$ in modo che il problema sia ben posto e il sistema abbia una e una sola soluzione.

Studiamo $\tilde{A}\tilde{x} = \tilde{b}$, vogliamo misurare errori in input e valutare l' errore che otteniamo in output di conseguenza, prendiamo solo il caso semplificato in cui solo il termine noto è perturbato (situazione comune in realtà), quindi $\tilde{b} \neq b, \tilde{A} = A$.

Definiamo alcuni vettori condensati poi in un singolo numero

- $\delta b := \tilde{b} - b \in \mathbb{R}^n$ (vettore di errori in input) $\rightarrow \|\delta b\|$ (errore assoluto)
- $\delta x := \tilde{x} - x \in \mathbb{R}^n$ (vettore di errori in output) $\rightarrow \|\delta x\|$ (errore assoluto)

Dal momento che per studiare il condizionamento di un problema abbiamo bisogno dei valori relativi definiamo:

- $\varepsilon_b := \frac{\|\delta b\|}{\|b\|}$ (errore relativo in input)
- $\varepsilon_x := \frac{\|\delta x\|}{\|x\|}$ (errore relativo in output)

Ci chiediamo adesso quale sia la relazione tra ε_x e ε_b pink Data una norma matriciale indotta $\forall A \in \mathbb{R}^{n \times n}$ con $\det(A) \neq 0, \forall b \in \mathbb{R}^n \rightarrow \varepsilon_x \leq \|A\| \cdot \|A^{-1}\| \varepsilon_b$. $\mu(A) = \|A\| \cdot \|A^{-1}\|$ è il **condizionamento della matrice**.

15.1 Dimostrazione

Ricordando che $\varepsilon_b := \frac{\|\delta b\|}{\|b\|}$ e $\varepsilon_x := \frac{\|\delta x\|}{\|x\|}$

Vogliamo maggiorare una frazione, quindi dobbiamo dimostrare che il numeratore e' minore di un certo valore e il denominatore e' maggiore di tale valore.

$$\|x\| >? \tag{5}$$

Sappiamo che la soluzione esatta risolve il sistema lineare, quindi $Ax = b$, quindi vale anche $\|Ax\| = \|b\|$, stiamo trattando una norma indotta possiamo applicare la quinta proprieta',scriviamo $\|Ax\| = \|b\| \leq \|A\| \cdot \|x\| \rightarrow \|x\| \geq \frac{\|b\|}{\|A\|}$.

Consideriamo la versione perturbata del sistema $A\tilde{x} = \tilde{b}$, sostituiamo \tilde{x}, \tilde{b} , con i valori definiti prima (SPECIFICARE BENE QUALI SONO), otteniamo quindi

$$A(x + \delta x) = b + \delta b$$

Ricordando che $Ax = b$, semplifichiamo

$$A\cancel{x} + A\delta x = \cancel{b} + \delta b$$

Dal momento che $\det(A) \neq 0$ possiamo moltiplicare entrambi i membri per A^{-1}

$$\underbrace{A^{-1}A}_{I} \delta x = A^{-1} \delta b \rightarrow \|\delta x\| = \|A^{-1} \cdot \delta b\| \leq \|A^{-1}\| \cdot \|\delta b\|$$

Possiamo adesso scrivere

$$\varepsilon_x = \frac{\|\delta x\|}{\|x\|} \leq \frac{\|A^{-1}\| \cdot \|\delta b\|}{\frac{\|b\|}{\|A\|}} = \|A^{-1}\| \cdot \|\delta b\| \cdot \frac{\|A\|}{\|b\|}$$

$$\mu(A) = \|A^{-1}\| \|A\|, \varepsilon_b = \frac{\|\delta b\|}{\|b\|}$$

Vediamo un esempio

$$\begin{cases} x + y = 2 \\ 1001x + 1000y = 2001 \end{cases} \quad A = \begin{pmatrix} 1 & 1 \\ 1001 & 1000 \end{pmatrix}, \|A\|_\infty = 2001$$

Svolgendo alcuni calcoli otteniamo che $A^{-1} = \begin{pmatrix} -1000 & 1 \\ 1001 & -1 \end{pmatrix}, \|A\|_\infty = 1002, \mu_\infty(A) = 2001 \cdot 1002 \approx 2 \cdot 10^6$

Questo significa che ogni errore in input viene moltiplicato per $2 \cdot 10^6$ (matrice malcondizionata).

Il valore migliore per il condizionamento di una matrice $\mu(A)$ e':

$$\underbrace{\|A \cdot A^{-1}\|}_{\|I\|=1 \text{ se } \|\cdot\| \text{ indotta}} \leq \|A\| \cdot \|A^{-1}\| \text{ (proprietà)} \rightarrow \mu(A) \geq 1$$

Alcune matrici note in letteratura:

15.2 Matrice di Hilbert

$$H_5 = \begin{bmatrix} 1 & \frac{1}{2} & \frac{1}{3} & \frac{1}{4} & \frac{1}{5} \\ \frac{1}{2} & \frac{1}{3} & \frac{1}{4} & \frac{1}{5} & \frac{1}{6} \\ \frac{1}{3} & \frac{1}{4} & \frac{1}{5} & \frac{1}{6} & \frac{1}{7} \\ \frac{1}{4} & \frac{1}{5} & \frac{1}{6} & \frac{1}{7} & \frac{1}{8} \\ \frac{1}{5} & \frac{1}{6} & \frac{1}{7} & \frac{1}{8} & \frac{1}{9} \end{bmatrix}$$

15.3 Matrice di Wilkinson

$$W_5 = \begin{bmatrix} 2 & 1 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 2 \end{bmatrix}$$

15.4 Matrice di Lehmer

$$\text{Data } A = (a_{ij}, a_{ij} = \begin{cases} \frac{i}{j} & j \geq i \\ \frac{j}{i} & j < i \end{cases}$$

Verifichiamo con Octave (LINK) quanto affermato fino ad adesso
Inganniamo, definiamo la soluzione voluta e otteniamo il termine noto.

```
>> H=hilb(6)
H =
    1.0000    0.5000    0.3333    0.2500    0.2000    0.1667
    0.5000    0.3333    0.2500    0.2000    0.1667    0.1429
    0.3333    0.2500    0.2000    0.1667    0.1429    0.1250
    0.2500    0.2000    0.1667    0.1429    0.1250    0.1111
    0.2000    0.1667    0.1429    0.1250    0.1111    0.1000
    0.1667    0.1429    0.1250    0.1111    0.1000    0.0909

>> x=ones(6,1)
x =
    1
    1
    1
    1
    1
```

```

1
1
>> b=H*x
b =
2.4500
1.5929
1.2179
0.9956
0.8456
0.9058
0.1270
0.9134
0.6324
0.0975

```

Vediamo adesso la soluzione esatta:

```

>> xx=H\b
xx =
0.9975
1.0636
0.6049
1.9624
-0.0084
1.3808

```

La soluzione esatta inizia ad allontanarsi molto da 1 (condizionamento molto importante). Quantifichiamo quanto osservato "ad occhio", misuriamo ε_b

```

>> eb=norm(db)/ norm(b)
eb = 4.7287e-07

```

e adesso ε_x

```

>> ex=norm(xx-x)\norm(x)
ex = 0.6121

```

Verifichiamo il condizionamento della matrice

```

>> cond(H)
ans = 1.4951e+07

```

L' errore peggiora allo scalare della dimensione della matrice:

```

>> for n=1:20
H = hilb(n);
muH(n)=cond(H);
end
>> muH

```

```

muH =
    1.0e+18 *
Columns 1 through 9
    0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000
Columns 10 through 18
    0.0000 0.0005 0.0162 0.4786 0.2551 0.2496 0.4887 0.4514 1.3500
Columns 19 through 20
    1.2579 2.1065

```

Matlab scala automaticamente all'ordine di grandezza maggiore, in questo caso 10^{18} , per potere vedere i rimi valori conviene impostare un formato in notazione scientifica `format short e`.

```

>> muH
muH =
Columns 1 through 7
    1.0000e+00 1.9281e+01 5.2406e+02 1.5514e+04 4.7661e+05 1.4951e+07 4.7537e+08
Columns 8 through 14
    1.5258e+10 4.9315e+11 1.6025e+13 5.2202e+14 1.6212e+16 4.7864e+17 2.5515e+17
Columns 15 through 20
    2.4960e+17 4.8875e+17 4.5144e+17 1.3500e+18 1.2579e+18 2.1065e+18

```

Plottiamo un grafico usando `plot(muH)`

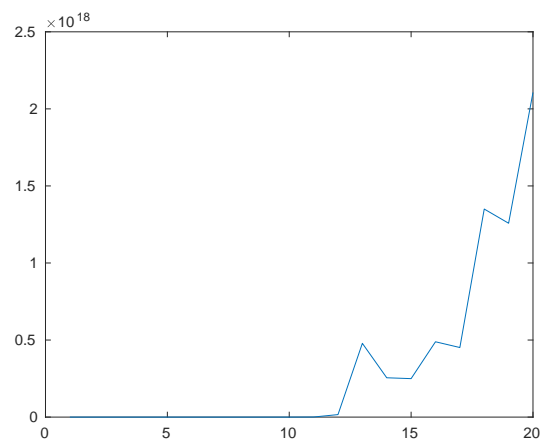


Figure 1: Andamento del condizionamento della matrice di Hilbert

Anche questo e' poco indicativo, cambiamo scala usando `semilogy(muH)`

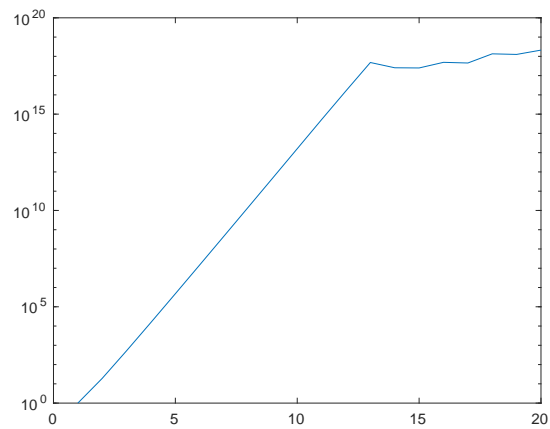


Figure 2: Andamento del condizionamento della matrice di Hilbert

Ora possiamo apprezzare la crescita esponenziale dell' andamento del condizionamento.

Nella seconda parte del grafico la crescita non e' piu' esponenziale a causa di errori algoritmici commessi nel calcolo del condizionamento della matrice.

Creiamo un vettore da usare per le matrici di Wilkinson e di Lehmer:

```
>> dim=10:10:200
```

Otteniamo il condizionamento e plottiamolo:

```
for n=dim
    W=wilkinson(n);
    muW(n)=cond(W);
end;
```

```
plot(dim,muW(dim))
```

L' andamento del condizionamento della matrice di Wilkinson e' perfettamente lineare

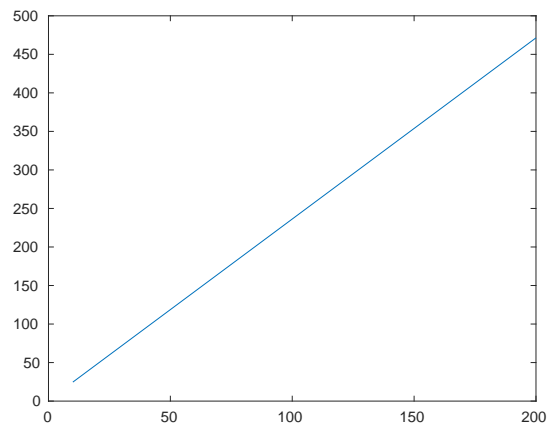


Figure 3: Andamento del condizionamento per la matrice di Wilkinson

```
>> dim=10:10:200
```

Otteniamo il condizionamento e plottiamolo:

```
for n=dim  
    L=gallery('lehmer',n);
```

```
muL(n)=cond(L);  
end;
```

```
plot(dim,muL(dim))
```

L' andamento del condizionamento della matrice di Lehmer e' poco piu' che lineare

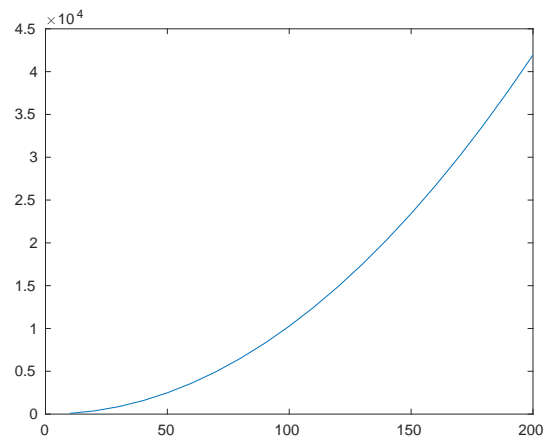


Figure 4: Andamento del condizionamento per la matrice di Lehmer

Per stabilire quale sia la crescita di tipo polinomiale $y \approx cn^\lambda$ possiamo estrarre il logaritmo, ottenendo $\log y \approx \log c + \lambda \log n$, nel momento in cui mettiamo entrambi gli assi in scala logaritmica otteniamo una retta che ha come coefficiente angolare λ , forziamo entrambi gli assi in scala logaritmica:

`loglog(dim,muA(dim))`

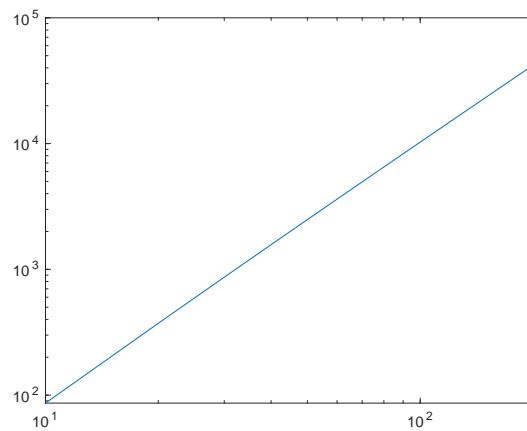


Figure 5: Andamento del condizionamento per la matrice di Lehmer in scala logaritmica

Confrontando con altri grafici possiamo determinare che la crescita del condizionamento per la matrice di Lehmer sia molto vicino ad una crescita quadratica

```
loglog(dim,muL(dim),'b', dim,dim.^2, 'r', dim,dim.^3, 'g')
```

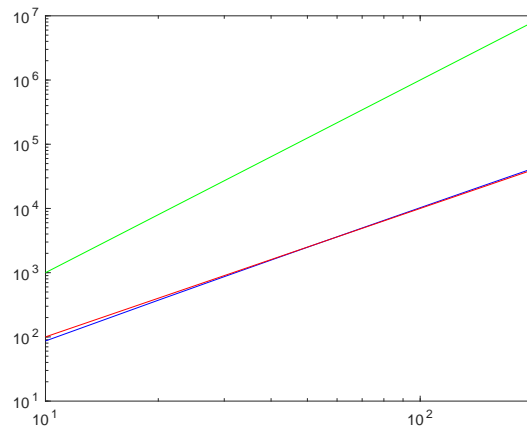


Figure 6: Andamento del condizionamento per la matrice di Lehmer in scala logaritmica

16 Collassamento sull' origine

$$A = 0 \in \mathbb{R}^{m \times n} x \in \mathbb{R}^n \rightsquigarrow A \cdot x = 0 \cdot x = 0 \in \mathbb{R}^m$$

17 Trasformazione identica (lascia tutto fermo)

$$A = I \in \mathbb{R}^{n \times n} x \in \mathbb{R}^n \rightsquigarrow A \cdot x = I \cdot x = x \in \mathbb{R}^n$$

18 s

$A = 2I = \begin{pmatrix} 2 & 0 \\ 0 & 2 \end{pmatrix} \in \mathbb{R}^{n \times n} x \in \mathbb{R}^n \rightsquigarrow A \cdot x = 2I \cdot x = 2x \in \mathbb{R}^n$ (geometricamente raddoppio della lunghezza di un vettore, in matematica **omotetia**) (dilatazione)

$A = \frac{1}{2}I = \begin{pmatrix} \frac{1}{2} & 0 \\ 0 & \frac{1}{2} \end{pmatrix} \in \mathbb{R}^{n \times n} x \in \mathbb{R}^n \rightsquigarrow A \cdot x = \frac{1}{2}I \cdot x = \frac{1}{2}x \in \mathbb{R}^n$ (geometricamente dimezzare la lunghezza di un vettore, sempre una **omotetia** (contrazione)).

19 matrice uguale meno id

$A = -I = \begin{pmatrix} -1 & 0 \\ 0 & -1 \end{pmatrix} \in \mathbb{R}^{n \times n} x \in \mathbb{R}^n \rightsquigarrow A \cdot x = -I \cdot x = -x \in \mathbb{R}^n$ (geometricamente simmetria rispetto all'origine)

$A = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix} \in \mathbb{R}^{2 \times 3} : \mathbb{R}^3 \rightarrow \mathbb{R}^2, x \in \mathbb{R}^3 (x_1/x_2/x_3) \rightarrow A \cdot x = \begin{pmatrix} x_1 \\ x_2 \end{pmatrix}$, (equivalente alla proiezione di vettori in 3D in 2D)

Ricordiamo che una matrice vista come trasformazione mappa vettori nello spazio delle colonne in vettori nello spazio delle righe

$B = \begin{pmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 0 \end{pmatrix} \in \mathbb{R}^{3 \times 2} \mathbb{R}^2 \rightarrow \mathbb{R}^3 \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} \rightarrow B \cdot x = \begin{pmatrix} x_1 \\ x_2 \\ 0 \end{pmatrix}$, (equivalente a portare in 3D un vettore in 2D, in matematica si chiama **immersione**)

Proprietà geometriche da studiare:

- **suriettività**: tutti i vettori di \mathbb{R}^m sono output di qualche vettore in \mathbb{R}^n
- **iniettività**: input diversi \rightarrow output diversi

20 Interpretazioni geometriche delle operazioni algebriche

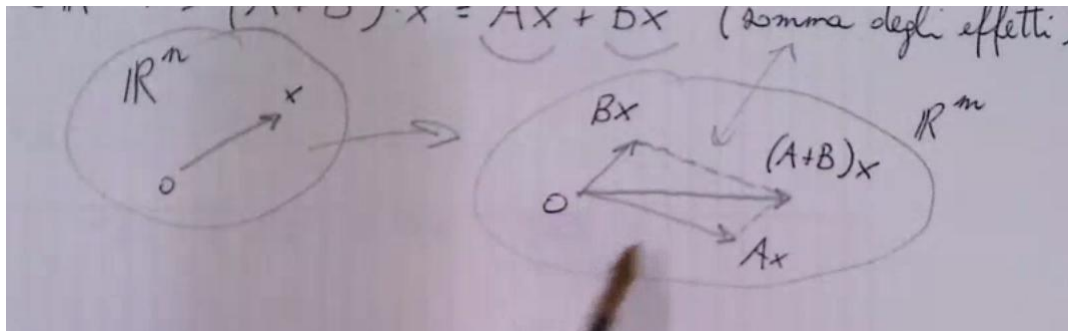
20.1 Somma tra Matrici

$A+B \in \mathbb{R}^{m \times n}$ consideriamo un input $x \in \mathbb{R}^n \rightarrow (A+B)x = Ax+Bx$, la somma di matrici corrisponde alla somma degli effetti delle singole trasformazioni, diamo un significato geometrico alla parola "somma"

Questa operazione fa perdere di vista i singoli effetti, si dice che la somma ha un effetto di **interferenza**.

20.2 Scalatura di una matrice

Data una matrice $A \in \mathbb{R}^m \times n: x \in \mathbb{R}^n \rightarrow (\alpha A) \cdot x = \alpha(Ax) \in \mathbb{R}^m$, l'effetto di questa trasformazione è la scalatura degli effetti.

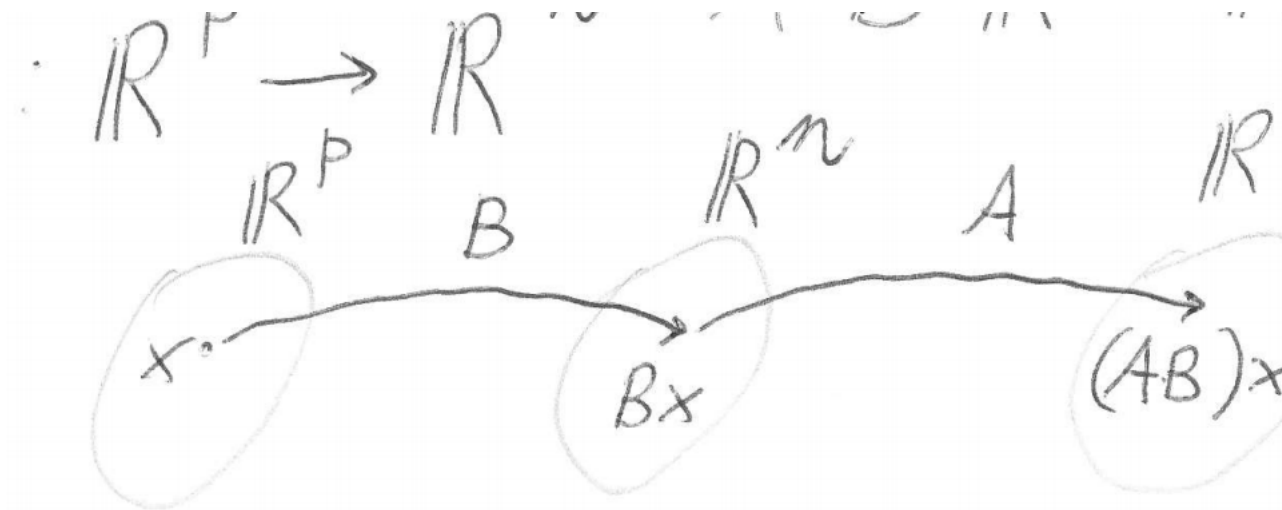


20.3 Prodotto righe per colonne

$A \in \mathbb{R}^{m \times n}, B \in \mathbb{R}^{n \times p}, A \cdot B \in \mathbb{R}^{m \times p}$ (NON COMMUTATIVO), ricordiamo che:

- $A : \mathbb{R}^n \rightarrow \mathbb{R}^m$
- $B : \mathbb{R}^p \rightarrow \mathbb{R}^n$
- $A \cdot B : \mathbb{R}^p \rightarrow \mathbb{R}^m$

Per studiare l'effetto $x \in \mathbb{R}^p \rightarrow (A+B)x = A(Bx)$ (mettiamo in evidenza la trasformazione di destra e poi moltiplichiamo per il primo fattore), ci stiamo muovendo in 3 spazi diversi



Osservazione: noi scriviamo il prodotto da sinistra verso destra, ma le trasformazioni agiscono nell'ordine contrario

per esempio

$$A = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix}, B = \begin{pmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 0 \end{pmatrix}, A \text{ rappresenta la "foto dall' alto", mentre}$$

B rappresenta l' "immersione".

$A \cdot B \rightarrow$ immersione \rightarrow foto dall' alto rappresenta l' identita in \mathbb{R}^2 .

$$\text{algebricamente } A \cdot B = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$$

$B \cdot A \rightarrow$ foto dall' alto \rightarrow immersione rappresenta la proiezione sulle altezze nulle in \mathbb{R}^3 . algebricamente $B \cdot A = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{pmatrix}$, questa e' una matrice non

nota, verifichiamo quindi cosa succede se:

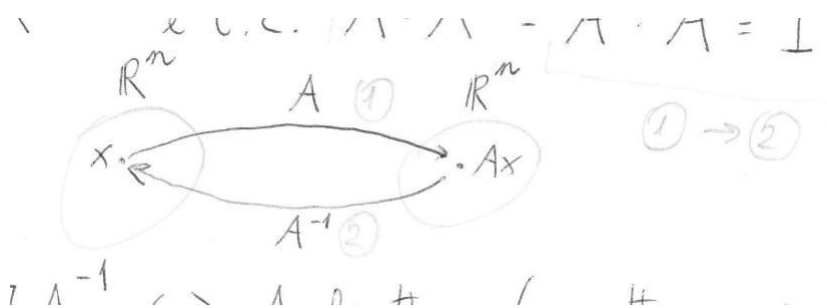
$$\begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} \rightarrow \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} = \begin{pmatrix} x_1 \\ x_2 \\ 0 \end{pmatrix}$$

Abbiamo dimostrato che corrisponde a quello che avevamo immaginato

21 Inversa della matrice

$A^{-1} \in \mathbb{R}^{n \times n}$ e' la matrice s.t. $\underbrace{A \cdot A^{-1}}_{1 \rightarrow 2} = \underbrace{A^{-1} \cdot A}_{2 \rightarrow 1} = I$, rappresenta la trasfor-

mazione inversa (output di A \rightarrow input), la possiamo fare se e solo se la trasfor-
mazione in considerazione e' biettiva, algebricamente sappiamo che se esiste la
matrice inversa A^{-1} allora $\det(A) \neq 0$



Per potere trattare questo argomento e' necessario prima definire alcune cose

22 Nucleo e Immagine

Il **nucleo** di una matrice:

$$\text{Ker}(A) := \{x \in \mathbb{R}^n : A \cdot x = 0 \in \mathbb{R}^n\}$$

Immagine di una matrice (insieme di tutti i possibili output)

$$R(A) := \{y \in \mathbb{R}^m : \exists x \in \mathbb{R}^n y = Ax\} \subseteq \mathbb{R}^m$$

Si puo' dimostrare che $\text{Ker}(A), R(A)$ sono sottospazi (ovvero dati $v, w \in \text{Ker}(A) \Rightarrow v + w, \alpha v \in \text{Ker}(A)$ e $v, w \in R(A) \Rightarrow v + w, \alpha v \in R(A)$) allora sia il Nucleo, sia l' Immagine hanno una base e una dimensione.

Osservazione

Data una matrice A dire che e' suriettiva equivale a dire che $R(A) = \mathbb{R}^m$
nucleo legato all' iniettivita

Theorem 22.1

Data una matrice A dire che e' iniettiva equivale a dire che $\text{Ker}(A) = \{0\}$
("nucleo banale")

Proof 22.1

\Rightarrow se A e' iniettiva considero $x \neq 0 \in \mathbb{R}^n$, l' iniettivita' implica che gli output siano diversi, quindi $Ax \neq A \cdot 0 = 0 \Rightarrow \text{Ker}(A) = \{0\}$

\Leftarrow se $\text{Ker}(A) = \{0\}$ vogliamo dimostrare che A e' iniettiva ($v \neq w \in \mathbb{R}^n \Rightarrow Av \neq Aw$). Se per assurdo $Av = Aw$ con $v \neq w$ allora $Av - Aw = 0 \Rightarrow A(v - w) = 0 \rightsquigarrow v - w \in \text{Ker}(A) = \{0\} \rightsquigarrow v - w = 0 \rightsquigarrow v = w$ (assurdo)

□

Ricapitolando:

- A surgettiva $\Leftrightarrow \dim R(A) = m$
- A iniettiva $\Leftrightarrow \dim \text{Ker}(A) = 0$

Definizione importante: Rango e caratteristica di A $\text{rk}(A) = \dim R(A)$, quindi se A e' suriettiva $\text{rk}(A) = m$

Theorem 22.2

$R(A) = \langle Ae_1, Ae_2, \dots, Ae_n \rangle$ dove e_j e' il vettore canonico che ha 1 in posizione j e zero in ogni altra posizione, per esempio $e_2 = \begin{pmatrix} 0 \\ 1 \\ \vdots \\ 0 \end{pmatrix} \in \mathbb{R}^n$

osservazione: se moltiplichiamo una matrice per un vettore canonico j otteniamo la j -esima colonna

$$Ae_j = \begin{pmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ \vdots & & & \\ a_{n1} & \dots & \dots & a_{nn} \end{pmatrix} \begin{pmatrix} 0 \\ \vdots \\ 1 \\ \vdots \\ 0 \end{pmatrix} = \begin{pmatrix} a_{1j} \\ \vdots \\ a_{mj} \end{pmatrix} \in \mathbb{R}^m$$

Quindi tutte i valori che corrispondono alla combinazione lineare precedente sono le colonne di A .

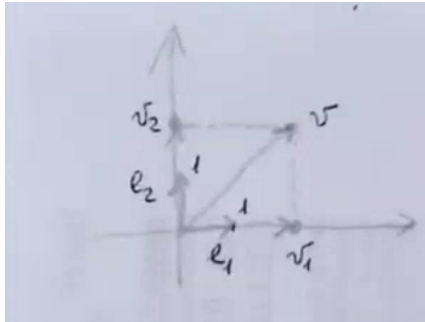
Inoltre se consideriamo il vettore canonico e_i^t ovvero il vettore di tutti zeri e uno in posizione i -esima $e_i^t A =$ riga i -esima di A , se combiniamo

$$e_i^t Ae_j = a_{ij}$$

possiamo quindi staccare elementi da una matrice.

Proof 22.2: g

Geometricamente possiamo rappresentare un generico vettore v come:



$$v = v_1 e_1 + v_2 e_2$$

Piu' genericamente,

dato $x = x_1 e_1 + x_2 e_2 + \dots + x_n e_n$ possiamo scrivere:

$$\begin{aligned} x \in \mathbb{R}^n &\rightarrow Ax \in R(A) \\ \Rightarrow Ax &= A(x_1 e_1 + x_2 e_2 + \dots + x_n e_n) = \\ &= x_1 (Ae_1) + x_2 (Ae_2) + \dots + x_n A(e_n) \\ &\in \langle Ae_1, \dots, Ae_n \rangle \end{aligned}$$

□

Osservazione: Dal momento che $R(A) \subseteq \mathbb{R}^m$, $R(A)$ e' generata da n colonne di $A \Rightarrow rk(A) = dim(R(A)) \leq \min(m, n)$

(avevamo gia' detto con la rossi che il rango e' maggiorato dal minimo delle dimensioni)

23 Nuclei e Immagini di matrici viste

- $A = O(x \rightarrow o\forall x)$ e' facile osservare che $Ker(A) = \mathbb{R}^n$, $R(A) = \{0\}$ inoltre $rk(A) = 0$ non e' iniettiva perche' molta roba nel nucleo non suriettiva

perche' l' immagine non e' \mathbb{R}^n

- $A = I \in \mathbb{R}^{n \times n} (x \rightarrow x)$, osserviamo che $Ker(A) = \{0\}$, $R(A) = \mathbb{R}^n$ inoltre $rk(A) = n$ sia iniettiva che suriettiva, l' identita' e' quindi una trasformazione biunivoca

- $A = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix} (\mathbb{R}^3) \rightarrow \mathbb{R}^2$

$Ker(A) = \{x \in \mathbb{R}^3 \rightarrow Ax = 0\}$, interpretando algebricamente il nucleo e' rappresentato dalle soluzioni del sistema lineare omogeneo (in cui x_3 non compare), quindi $\begin{cases} x_1 = 0 \\ x_2 = 0 \end{cases} \rightarrow Ker(A) = \left\{ \begin{pmatrix} 0 \\ 0 \\ x_3 \end{pmatrix} \mid x_3 \in \mathbb{R} \right\} = \langle e_3 \rangle$

in pratica ogni volta che vogliamo calcolare il nucleo troviamo le soluzioni del sistema lineare omogeneo assegnato alla matrice

$$R(A) = \mathbb{R}^2 \text{ (matrice suriettiva ma non iniettiva)}$$

- $A = \begin{pmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 0 \end{pmatrix} (\mathbb{R}^2 \rightarrow \mathbb{R}^3)$ (immersione)

$Ker(A) = \{0\}$ "ragionando geometricamente", si puo' ottenere algebricamente risolvendo il sistema omogeneo (verificata iniettivita')

$R(A) = \{\text{altezze nulle}\} = \left\{ \begin{pmatrix} x_1 \\ x_2 \\ 0 \end{pmatrix} \mid x_1, x_2 \in \mathbb{R} \right\} = \langle e_1, e_2 \rangle$ l' immagine non coincide con \mathbb{R}^3 quindi non e' suriettiva

Theorem 23.1

Si puo' dimostrare che $\forall A \in \mathbb{R}^{m \times n} \dim Ker(A) + rk(A) = n$ Ne segue che:

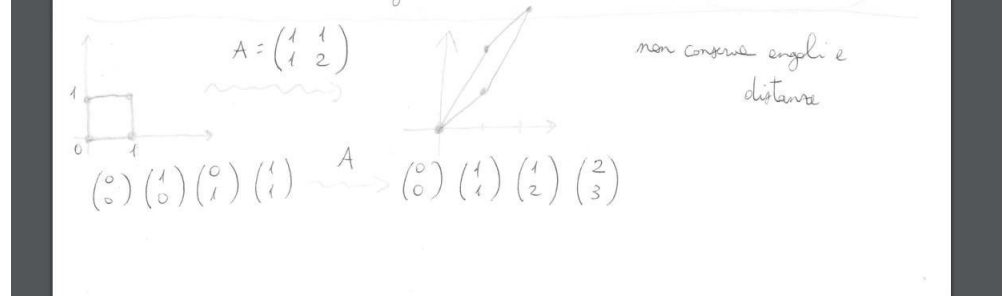
- Una matrice e' iniettiva se e solo se $Ker(A) = \{0\}$ oppure se $rk(A) = n$
- Se $m < n \Rightarrow rk(A) \leq \min(m,n) = m < n$ (ovvero non puo' mai succedere quanto detto al punto precedente), una matrice con $m < n$ non sara' mai iniettiva
- $m = n$ A iniettiva se e solo se $Ker(A) = \{0\} \Leftrightarrow rk(A) = n$ quindi la matrice e' anche suriettiva, quindi la matrice e' biettiva, invertibile e ha determinante non nullo. Quindi per capire se una matrice quadrata e' iniettiva/suriettiva e' sufficiente calcolare il determinante

24 Quindi

Una matrice e' iniettiva se e solo se $Ker(A) = \{0\}$ oppure se $rk(A) = n$

Facciamo ora delle trasformazioni e vediamo cosa succede

Disegniamo il quadrato che ha vertici $(0,0), (1,0), (0,1), (1,1)$



Moltiplicando i vettori per la matrice $A = \begin{pmatrix} 1 & 1 \\ 1 & 2 \end{pmatrix}$ otteniamo il secondo grafico, poco interessante, non conserva ne' angoli ne distanze.

Consideriamo ora la matrice $A = \begin{pmatrix} 1 & 1 \\ 1 & 1 \end{pmatrix}$

$$\begin{pmatrix} 1 & 1 \\ 1 & 1 \end{pmatrix} \begin{pmatrix} 0 \\ 0 \end{pmatrix}, \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ 1 \end{pmatrix}, \begin{pmatrix} 1 \\ 1 \end{pmatrix} \rightarrow \begin{pmatrix} 0 \\ 0 \end{pmatrix}, \begin{pmatrix} 1 \\ 1 \end{pmatrix}, \begin{pmatrix} 1 \\ -1 \end{pmatrix}, \begin{pmatrix} 2 \\ 0 \end{pmatrix}$$

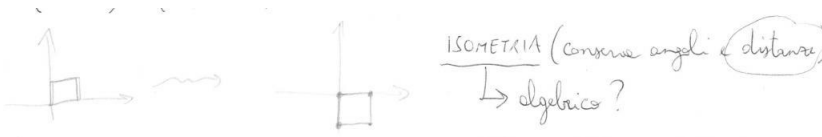


Questa trasformazione

conserva gli angoli ma non le distanze

Consideriamo ora la matrice $A = \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix}$

$$\begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix} \begin{pmatrix} 0 \\ 0 \end{pmatrix}, \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ 1 \end{pmatrix}, \begin{pmatrix} 1 \\ 1 \end{pmatrix} \rightarrow \begin{pmatrix} 0 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ -1 \end{pmatrix}, \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \begin{pmatrix} 1 \\ -1 \end{pmatrix}$$



Questa

trasformazione e' una isometria, ovvero che conserva sia angoli e distanze

Le **isometrie** sono le trasformazioni piu' interessanti, vediamo di caratterizzarle algebricamente

25 Isometrie

Trattiamo solamente le matrici quadrate ($A \in \mathbb{R}^{n \times n}$) Ci concentriamo sul conservare tutte le distanze (implica il conservare anche gli angoli).

Immaginiamo di avere $x, y \in \mathbb{R}^n$, la distanza tra i punti e' $\|x - y\|$ (input), vogliamo che in input succeda la stessa cosa, quindi $Ax, Ay \in \mathbb{R}^n$ la distanza tra i due punti adesso e' $\|Ax - Ay\|$, dal momento che la trasformazione e' lineare possiamo anche scrivere la distanza come $\|A(x - y)\|$. A e' una isometria se per ogni differenza v la nuova differenza coincide, ovvero

$$\forall v \in \mathbb{R}^n \|A_v\| = \|v\|$$

Consideriamo adesso una classe di matrici: le **matrici ortogonali**.

Una matrice si dice **ortogonale** se $A^t A = A A^t = I$. Dal momento che moltiplicare una matrice e ottenere l'identità equivale a invertire la matrice ($A^{-1} = A^t$).

Osserviamo che se consideriamo $A^t A = I$, la riga i -esima e la j -esima colonna del prodotto lo possiamo scrivere come:

$$\begin{aligned} & (\text{i-esima riga di } A^t) \times (\text{j-esima colonna di } A) = I_{ij} \\ & = (\text{i-esima colonna di } A)^t \times (\text{j-esima colonna di } A) \end{aligned}$$

Questo coincide con il prodotto scalare tra colonne i e j di A , quindi in base alla definizione di matrice Identica A è una matrice ortogonale se e solo se ha colonne ortonormali (ovvero ortogonali e di lunghezza 1).

Esempio $\begin{pmatrix} 1 & 1 \\ 1 & 2 \end{pmatrix}$ non è ortogonale perché la prima e la seconda colonna non sono ortogonali, invece $\begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}$ non è ortogonale, le colonne tra di loro sono ortogonali ma non ortonormali.

$$\left\| \begin{pmatrix} 1 \\ 1 \end{pmatrix} \right\|_2 = \sqrt{2}, \left\| \begin{pmatrix} 1 \\ -1 \end{pmatrix} \right\|_2 = \sqrt{2}$$

Se ora dividiamo ogni colonna per il suo modulo chiaramente non perdiamo la ortogonalità, ma otteniamo una matrice ortogonale facendo la normalizzazione delle colonne

$$\left\| \begin{pmatrix} \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} \end{pmatrix} \right\|_2 = 1, \left\| \begin{pmatrix} \frac{1}{\sqrt{2}} \\ -\frac{1}{\sqrt{2}} \end{pmatrix} \right\|_2 = 1$$

Un esempio di matrice ortogonale è quindi

$$\begin{pmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} \end{pmatrix}$$

È una matrice ortogonale anche $\begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix}$ (usata precedentemente nell'esempio della isometria)

Theorem 25.1

Una matrice quadrata è ortogonale se e solo se è un'isometria, ovvero:

$$A \in \mathbb{R}^{n \times n} \text{ ortogonale } (A A^t = I) \Leftrightarrow \text{isometria } (\forall v \in \mathbb{R}^n \|A v\| = \|v\|)$$

corollario

se abbiamo due matrici ortogonali A, B anche A per B è ortogonale (composizione di isometrie) e anche $A^{-1} = A^t$

Proof 25.1

\Rightarrow Hp: $AA^t = I$, Th: $\|A_v\| = \|v\|$, e' piu' comodo elevare al quadrato le norme perche' la norma al quadrato di un vettore si puo' sempre scrivere come prodotto scalare del vettore con se stesso ($v^t v = v_1^2 + v_2^2 + \dots + v_n^2$)

$$\begin{aligned}\|A_v\|^2 &= (Av)^t(Av) \\ &= v^t A^t Av \\ &= v^t I v \\ &= v^t v = \|v\|^2\end{aligned}$$

□

\Leftarrow Hp $\|A_v\| = \|v\|$ in particolare vale per i vettori canonici $\forall j \|Ae_j\| = \|e_j\| = 1$, quindi ogni colonna di A ha lunghezza 1, dal momento che l' ipotesi e' la definizione di isometria sappiamo che l' angolo tra due vettori viene preservato, prendiamo quindi i vettori canonici generici e_i e $e_j (i \neq j)$, per definizione $e_i \perp e_j$ quindi Ae_i, Ae_j sono ortogonali (due colonne di A sono ortogonali), queste due proprieta' corrisponondono ad essere una **matrice ortogonale** (ipotesi)

□

Esempio matrice di permutazione 5*5 ($e_3 e_5 e_1 e_4 e_2$), ovvero $\Pi = \begin{pmatrix} 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 \end{pmatrix}$

La matrice e' chiaramente ortogonale, vediamo cosa succede se calcoliamo $\Pi \begin{pmatrix} v_1 \\ v_2 \\ v_3 \\ v_4 \\ v_5 \end{pmatrix} =$

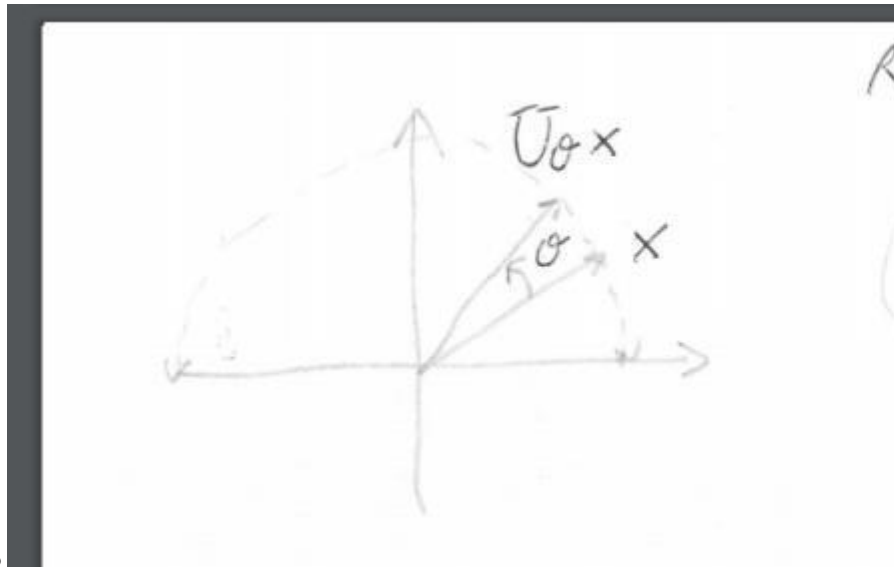
$\begin{pmatrix} v_3 \\ v_5 \\ v_1 \\ v_4 \\ v_2 \end{pmatrix}$ Ovvero prende elementi di un vettore e ne cambia l' ordine.

Theorem 25.2

Facendo la riduzione di Gauss con pivoting parziale otteniamo $\Pi A = LU$

26 Esempi notevoli di isometrie

rotazioni e simmetrie



Rotazioni nel piano cartesiano

Scriviamo la matrice ottenuta facendo la rotazione dei versori $e_1 = \begin{pmatrix} 1 \\ 0 \end{pmatrix}$ e

$$e_2 = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$$

$$U_\Theta = \begin{pmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{pmatrix}$$

Vediamo se la matrice e' ortogonale, ovvero se $U_\Theta = U_\Theta^t$

$$\begin{aligned} & \begin{pmatrix} \cos \theta & \sin \theta \\ -\sin \theta & \cos \theta \end{pmatrix} \begin{pmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{pmatrix} = \\ & = \begin{pmatrix} \cos^2 \theta + \sin^2 \theta & -\cos \theta \sin \theta + \sin \theta \cos \theta \\ -\sin \theta \cos \theta + \cos \theta \sin \theta & \sin^2 \theta + \cos^2 \theta \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \end{aligned}$$

Esercizio

Dato $x = \begin{pmatrix} x_1 \\ x_2 \end{pmatrix}$ trovare U_θ s.t. $U_\theta x = \begin{pmatrix} \alpha \\ 0 \end{pmatrix}$ per un certo α (algoritmo di azzeramento).

Per prima cosa osserviamo che la lunghezza del vettore $\begin{pmatrix} \alpha \\ 0 \end{pmatrix}$, ovvero $\|\begin{pmatrix} \alpha \\ 0 \end{pmatrix}\|_2 = |\alpha|$, dal momento che stiamo trattando una isometria la lunghezza in input deve essere uguale alla lunghezza in output, quindi:

$$\|x\|_2 = \sqrt{x_1^2 + x_2^2} = \|\begin{pmatrix} \alpha \\ 0 \end{pmatrix}\|_2 = |\alpha| \rightsquigarrow \alpha = \pm \|x\|_2$$

Ora calcoliamo il prodotto $U_\theta \cdot x$

$$U_\theta \cdot x = \begin{pmatrix} x_1 \cos \theta & -x_2 \sin \theta \\ x_1 \sin \theta & x_2 \cos \theta \end{pmatrix}$$

Confrontando la matrice ottenuta con il dato:

$$\begin{aligned} x_1 \sin \theta + x_2 \cos \theta &= 0 \\ \frac{\sin \theta}{\cos \theta} &= -\frac{x_2}{x_1} \\ \tan \theta &= -\frac{x_2}{x_1} \end{aligned}$$

Ricordando che:

$$\cos \theta = \frac{1}{\sqrt{1 + \tan^2 \theta}} \text{ e } \sin \theta = \frac{\tan \theta}{\sqrt{1 + \tan^2 \theta}}$$

Scriviamo quindi:

$$\cos \theta = \frac{x_1}{\sqrt{x_1^2 + x_2^2}} \text{ e } \sin \theta = \frac{-x_2}{\sqrt{x_1^2 + x_2^2}}$$

Estendiamo questo ragionamento in \mathbb{R}^n

27 Rotazioni di Givens

Le rotazioni di Givens sono rotazioni fatte sul piano di vettori coordinati, le indichiamo con $G(i, j, \theta)$, corrisponde a una rotazione piana U_θ nel "piano" generato dai vettori canonici $\langle e_i, e_j \rangle$ e assomiglia alla trasformazione identica nelle altre direzioni. Per ruotare nel piano orizzontale scegliamo gli indici 1 e 2. Definiamo quindi:

- $G_{ii} = G_{jj} = \cos \theta := c$
- $G_{ij} = \sin \theta := s$
- $G_{ji} = -s$

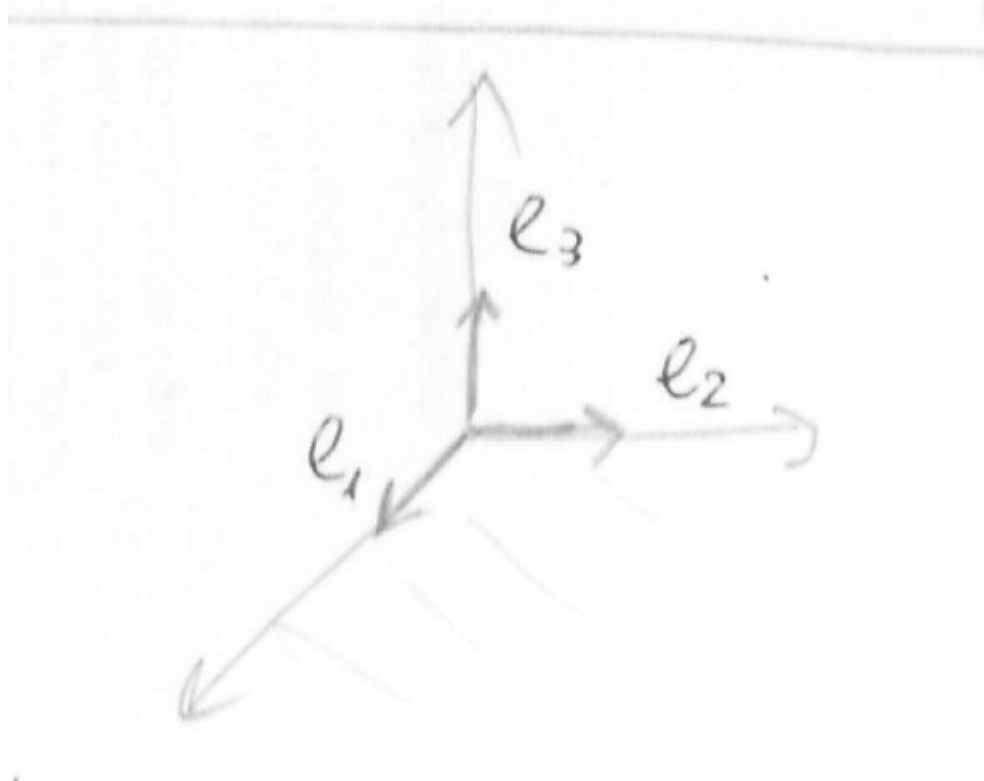


Figure 7: Esempio di Rotazione di Givens in \mathbb{R}^3

Per esempio se $i < j$ $G(i, j, \theta)$ scriviamo nelle posizioni precedentemente definite i valori indicati e filliamo con gli elementi di una matrice identica (sulle altre componenti non deve cambiare nulla).

$$\begin{pmatrix} 1 & \cdots & 0 & \cdots & 0 & \cdots & 0 \\ \vdots & \ddots & \vdots & & \vdots & & \vdots \\ 0 & \cdots & c & \cdots & -s & \cdots & 0 \\ \vdots & & \vdots & \ddots & \vdots & & \vdots \\ 0 & \cdots & s & \cdots & c & \cdots & 0 \\ \vdots & & \vdots & & \vdots & \ddots & \vdots \\ 0 & \cdots & 0 & \cdots & 0 & \cdots & 1 \end{pmatrix}$$

Naturalmente se gli indici fossero rovesciati (ovvero $i > j$) la rotazione sarebbe:

$$\begin{pmatrix} 1 & \cdots & 0 & \cdots & 0 & \cdots & 0 \\ \vdots & \ddots & \vdots & & \vdots & & \vdots \\ 0 & \cdots & c & \cdots & s & \cdots & 0 \\ \vdots & & \vdots & \ddots & \vdots & & \vdots \\ 0 & \cdots & -s & \cdots & c & \cdots & 0 \\ \vdots & & \vdots & & \vdots & \ddots & \vdots \\ 0 & \cdots & 0 & \cdots & 0 & \cdots & 1 \end{pmatrix}$$

Moltiplichiamo una rotazione di Givens per un vettore arbitrario $x = \begin{pmatrix} x_1 \\ \vdots \\ x_n \end{pmatrix}$

$$G(i, j, \theta) \cdot x = \begin{pmatrix} x_1 \\ \vdots \\ x_{i-1} \\ cx_1 - sx_j \\ x_{i+1} \\ \vdots \\ x_{j-1} \\ sx_i + cx_j \\ \vdots \end{pmatrix}$$

Adesso possiamo scegliere un angolo s.t. $x \xrightarrow{G(i, j, \theta)}$

$$\begin{pmatrix} \square \\ \alpha \\ \square \\ 0 \\ \square \end{pmatrix} \begin{matrix} i \\ j \end{matrix}$$

(□ rappresenta i valori che rimangono invariati)

In una rotazione di Givens usiamo gli stessi valori ottenuti precedentemente per azzerare, in forma piu' generalizzata x_i e' il perno (l' unica componente che cambia) e x_j e' l' elemento da azzerare, quindi

- $c = \frac{x_i}{\sqrt{x_i^2 + x_j^2}}$

- $s = \frac{-x_j}{\sqrt{x_i^2 + x_j^2}}$

Allora $G(i, j, \theta) \cdot x = \begin{pmatrix} \square \\ \sqrt{x_i^2 + x_j^2} \\ \square \\ 0 \\ \square \end{pmatrix} \begin{matrix} i \\ j \end{matrix}$

Esercizio

$$x = \begin{pmatrix} 1 \\ -1 \\ 2 \\ 0 \\ 3 \end{pmatrix}$$

Se in x vogliamo azzerare 2 (elemento j) dobbiamo scegliere un valore da usare come pivot, in questo caso 1 .

Cerchiamo una rotazione di Givens $G(1, 3, \theta)$ s.t $x \rightarrow \begin{pmatrix} \alpha \\ -1 \\ 0 \\ 0 \\ 3 \end{pmatrix}$

Applichiamo le formule per determinare c ed s :

$$c = \frac{1}{\sqrt{1^2 + 2^2}} = \frac{1}{\sqrt{5}}$$
$$s = \frac{-2}{\sqrt{5}} = \frac{-2}{\sqrt{5}} \alpha = \sqrt{5}$$

Quindi $G(1, 3, \theta)$ s.t $x \rightarrow \begin{pmatrix} \alpha \\ -1 \\ 0 \\ 0 \\ 3 \end{pmatrix} = \begin{pmatrix} \sqrt{5} \\ -1 \\ 0 \\ 0 \\ 3 \end{pmatrix}$

$$G(1, 3, \theta) = \begin{pmatrix} c & 0 & -s & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ s & 0 & c & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix}$$

Geometricamente e' una rotazione antioraria nel piano $\langle e_1, e_3 \rangle$

Esercizio

Continuiamo al fine di azzerare ulteriori valori nel vettore ottenuto

Vogliamo azzerare -1 scegliendo come perno 3 $\begin{pmatrix} \sqrt{5} \\ -1 \\ 0 \\ 0 \\ 3 \end{pmatrix} \rightarrow \begin{pmatrix} \sqrt{5} \\ 0 \\ 0 \\ 0 \\ \alpha \end{pmatrix}$ Cer-

chiamo $G(5, 2, \theta)$

- $\alpha = \sqrt{3^2 + (-1)^2} = \sqrt{10}$
- $c = \frac{3}{\sqrt{10}}$
- $s = \frac{-(-1)}{\sqrt{10}} = \frac{1}{\sqrt{10}}$

Quindi $G(5, 2, \theta) = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & \frac{3}{\sqrt{10}} & 0 & 0 & \frac{1}{\sqrt{10}} \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & -\frac{1}{\sqrt{10}} & 0 & 0 & \frac{3}{\sqrt{10}} \end{pmatrix}$ Geometricamente e' una

rotazione antioraria nel piano $\langle e_5, e_2 \rangle$

Abbiamo ottenuto il vettore $\begin{pmatrix} \sqrt{5} \\ 0 \\ 0 \\ 0 \\ \sqrt{10} \end{pmatrix}$

Continuando riusciamo ad arrivare al vettore $\begin{pmatrix} \alpha \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}$ $\alpha = \sqrt{10} = \sqrt{5} = \sqrt{15}$

Geometricamente e' una rotazione antioraria nel piano $\langle e_1, e_5 \rangle$

Ricapitolando, a partire del vettore x abbiamo fatto:

$$G(1, 5, \theta_3) \left(G(5, 2, \theta_2) \left(G(1, 3, \theta_1) \left(\begin{pmatrix} 1 \\ -1 \\ 2 \\ 0 \\ 3 \end{pmatrix} \right) \right) \right) = \begin{pmatrix} \sqrt{15} \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}$$

Ricordiamo che geometricamente moltiplicare matrici corrisponde a comporre le trasformazioni da destra verso sinistra. Quello che otteniamo e' anche l'equivalente della moltiplicazione di tutto il prodotto delle matrici $5 * 5$ (composizione delle singole rotazioni), il prodotto $G(1, 5, \theta_3)G(5, 2, \theta_2)G(1, 3, \theta_1)$ e' una isometria, quindi come verifica dei calcoli fatti possiamo calcolare la lunghezza iniziale del vettore che deve essere uguale alla lunghezza del risultato,

quindi:

$$\|x\|_2 = \sqrt{1^2 + (-1)^2 + 2^2 + 3^2} = \sqrt{15} = \left\| \begin{pmatrix} \sqrt{15} \\ 0 \\ 0 \\ 0 \end{pmatrix} \right\|_2$$

Definiamo una abbreviazione $G(i, j, \theta) \rightarrow G_{ij}$

Per trovare un metodo alternativo alla riduzione di Gauss consideriamo una

intera matrice generica $A = \begin{pmatrix} x & x & x & x \\ x & x & x & x \\ x & x & x & x \\ x & x & x & x \\ x & x & x & x \end{pmatrix}$ Lavoriamo sulla prima colonna:

con una sequenza di rotazioni scegliamo sempre a_{11} come perno ed eliminiamo tutti i valori "sotto"

$$G_{16}G_{15}G_{14}G_{13}G_{12} \begin{pmatrix} x & x & x & x \\ x & x & x & x \\ x & x & x & x \\ x & x & x & x \\ x & x & x & x \\ x & x & x & x \end{pmatrix}$$

Quello che succede nelle colonne 2,3,4 corrisponde al prodotto delle rotazioni sulle singole colonne.

Una volta azzerata la prima colonna facciamo altre rotazioni in modo da ottenere una forma a scalini (come nella riduzione di Gauss), conviene considerare come pivot su a_{22} (scegliendo a_{21} la colonna 1 verrebbe modificata vanificando il lavoro precedente).

Aggiungiamo ora quindi

$$G_{26}G_{25}G_{24}G_{23}G_{16}G_{15}G_{14}G_{13}G_{12} \begin{pmatrix} x & x & x & x \\ x & x & x & x \\ x & x & x & x \\ x & x & x & x \\ x & x & x & x \\ x & x & x & x \end{pmatrix}$$

Per la terza colonna aggiungiamo

$$G_{36}G_{35}G_{34}G_{26}G_{25}G_{24}G_{23}G_{16}G_{15}G_{14}G_{13}G_{12} \begin{pmatrix} x & x & x & x \\ x & x & x & x \\ x & x & x & x \\ x & x & x & x \\ x & x & x & x \\ x & x & x & x \end{pmatrix}$$

E per la quarta

$$G_{46}G_{45}G_{36}G_{35}G_{34}G_{26}G_{25}G_{24}G_{23}G_{16}G_{15}G_{14}G_{13}G_{12} \begin{pmatrix} x & x & x & x \\ x & x & x & x \\ x & x & x & x \\ x & x & x & x \\ x & x & x & x \end{pmatrix}$$

Il risultato di questo procedimento, se la matrice e' del tipo $A \in \mathbb{R}^{m \times n}$ con $m \geq n$ e' la matrice composta da una sottomatrice triangolare superiore R di n righe, e sotto $m-n$ righe nulle. $G := G_{46}G_{45}G_{36}G_{35}G_{34}G_{26}G_{25}G_{24}G_{23}G_{16}G_{15}G_{14}G_{13}G_{12}$ sono tutte matrici $m \times m$ ed e' una composizione di rotazioni(isometrie), quindi e' una **matrice ortogonale** ($G^t G = I$, ovvero $G^t = G^{-1}$). Ricapitolando $G \cdot A = \begin{pmatrix} R \\ 0 \end{pmatrix} \rightarrow A = G^t \cdot \begin{pmatrix} R \\ 0 \end{pmatrix}$ Chiamata G^t Q (ortogonale) questa trasformazione prende il nome di **fattorizzazione QR** Q e' una matrice $m \times m$ mentre R e' una matrice quadrata $n \times n$.

Fattorizzazione LU (Gauss) costo per $m \times n$: $\frac{1}{2}n^2(m - \frac{n}{3})$ stabile solo con pivoting Parziale

Fattorizzazione QR (Givens) costo per $m \times n$: $2n^2(m - \frac{n}{3})$ (4 volte Gauss) sempre stabile

28 Proprieta' della fattorizzazione QR

Data la fattorizzazione QR: $A = Q \begin{pmatrix} R \\ 0 \end{pmatrix}$ dimostra che:

$A \in \mathbb{R}^{m \times n}, m \geq n, rk(A) = n$ (rango massimo) $\Rightarrow \det(R) \neq 0$, ovvero R e' invertibile e un eventuale sistema associato ha 1 e 1 sola soluzione

$$A \in \mathbb{R}^{m \times n}, m \geq n, rk(A) = n$$
(rango massimo) $\Rightarrow Q = \left(\underbrace{Q_0}_n \middle| * \right)_{m \times n} R(A) = R(Q_0)$,

dove $R(Q_0) = \langle \text{colonne di } Q_0 \rangle$, le colonne di una matrice Ortoogonale sono ortonormali ovvero sono ortogonali quindi prodotto scalare uguale a zero e hanno lunghezza 1 e $R(A) = \langle \text{colonne di } A \rangle$ (sappiamo che sono indipendenti per via dell' ipotesi sul rango) la fattorizzazione qr quindi calcola le basi ortonormali (come algoritmo di Gram-Schmidt ma meno costoso e piu' stabile)

Questo algoritmo e' troppo costoso, troviamo un modo alternativo di scrivere la fattorizzazione qr

29 Riflessioni di Householder

La matrice che definisce una **riflessione di Householder** e' $P = I - 2ww^t$ dove $w \in \mathbb{R}^n$ (e' un vettore colonna) con $\|w\| = 1$, quindi un versore.

Stiamo calcolando un prodotto colonna \times riga, otteniamo una matrice particolare che si chiama **diade**.

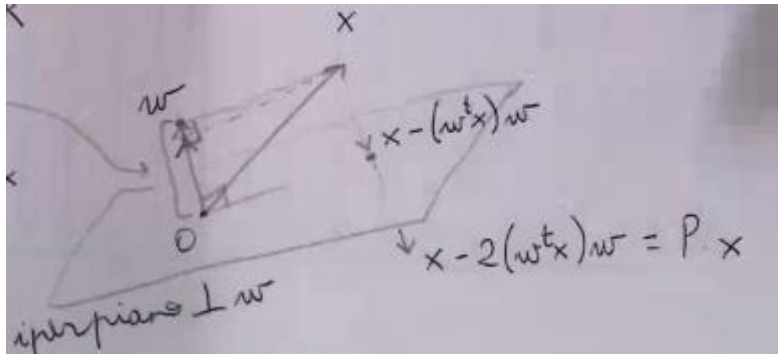


Figure 8: Rappresentazione geometrica della riflessione di Householder

Interpretazione geometrica (dato un vettore $x \in \mathbb{R}^n$ studiamo $P \cdot x$):

$$I \cdot x - 2ww^t \cdot x$$

$$I \cdot x - 2w(\underbrace{w^t x}_{\in \mathbb{R}})$$

$$x - 2 \underbrace{(w^t w)}_{\text{proiezione di } x \text{ lungo } w} w = P \cdot x$$

Geometricamente la riflessione di Householder corrisponde a una riflessione rispetto a un iperpiano.

Proprietà di queste matrici:

1. La matrice di Householder è simmetrica ($p = p^t$)
2. $p^2 = pp^t = I$ Geometricamente significa che se invertiamo l'operazione otteniamo nuovamente il valore iniziale

1.

$$p^t = (I - 2ww^t)^t =$$

$$= I^t - 2(ww^t)^t =$$

$$I - 2(w^t)^t w^t =$$

$$= I - 2ww^t = p$$

2.

$$(I - 2ww^t)(I - 2ww^t) =$$

$$= I \cdot I - 2ww^t \cdot I - I \cdot 2ww^t - 2ww^t \cdot (-2ww^t)$$

$$= I - 2ww^t - 2ww^t + 4w \underbrace{w^t w}_1 w^t = I$$

Esercizio

Dato $x \in \mathbb{R}^n$ cerchiamo $P = I - 2ww^t$ (quindi cerchiamo w ricordando

che $\|w\|_2 = 1$) s.t. $P \cdot x = \begin{pmatrix} \alpha \\ 0 \\ \vdots \\ 0 \end{pmatrix} = \alpha e_1$ Dal momento che P e' una

isometria $\alpha = \pm \|x\|_2$, scegliamo per esempio "+".

Consideriamo $u := x - \alpha e_1$, osserviamo che $u \perp w$, per ipotesi $\|w\|_2 = 1$, quindi w deve essere un versore di u , quindi $w = \frac{u}{\|u\|_2}$.

Dal punto di vista algebrico $P \cdot x = \alpha e_1$, che e' anche uguale a $x - 2(w^t x)w$, quindi $2(w^t x)w = x - \alpha e_1 = u$, estraendo i versori $w = \hat{u} = \frac{u}{\|u\|_2}$

Vediamo adesso un algoritmo (calcolo di P per azzerare x)

- $\alpha := \|x\|_2$
- $u := x - \alpha e_1$
- $w := \frac{u}{\|u\|_2}$
- $P := I - 2ww^t = I - 2 \frac{u}{\|u\|_2} \cdot \frac{u^t}{\|u\|_2} = I - 2 \frac{uu^t}{\|u\|_2^2} = I - 2 \frac{uu^t}{u^t u}$

Per esempio $x = \begin{pmatrix} 3 \\ 1 \\ 5 \\ 1 \end{pmatrix}$ vogliamo arrivare alla forma $\begin{pmatrix} \alpha \\ 0 \\ 0 \\ 0 \end{pmatrix}$

$$\alpha = \|x\|_2 = \sqrt{3^2 + 1^2 + 5^2 + 1^2} = \sqrt{36} = 6$$

$$u = x - 6e_1 = \begin{pmatrix} 3 \\ 1 \\ 5 \\ 1 \end{pmatrix} - \begin{pmatrix} 6 \\ 0 \\ 0 \\ 0 \end{pmatrix} = \begin{pmatrix} -3 \\ 1 \\ 5 \\ 1 \end{pmatrix}$$

$$u^t u = (-3 \ 1 \ 5 \ 1) \begin{pmatrix} -3 \\ 1 \\ 5 \\ 1 \end{pmatrix} = (-3)^2 + 1^2 + 5^2 + 1^2 = 36$$

$$P = I - 2 \frac{uu^t}{u^t u} = I - 2 \frac{uu^t}{36} = I - \frac{1}{18} uu^t$$

$$uu^t = \begin{pmatrix} -3 \\ 1 \\ 5 \\ 1 \end{pmatrix} (-3 \ 1 \ 5 \ 1) = \begin{pmatrix} 9 & -3 & -15 & 1 \\ -3 & 1 & 5 & 1 \\ -15 & 5 & 25 & 5 \\ -3 & 1 & 5 & 1 \end{pmatrix}$$

$$P = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} - \begin{pmatrix} \frac{1}{2} & -\frac{1}{6} & -\frac{5}{6} & -\frac{1}{6} \\ -\frac{1}{6} & \frac{1}{18} & \frac{5}{18} & \frac{1}{18} \\ -\frac{5}{6} & \frac{5}{18} & -\frac{5}{6} & -\frac{5}{18} \\ -\frac{1}{6} & \frac{1}{18} & \frac{5}{18} & -\frac{1}{18} \end{pmatrix} = \begin{pmatrix} \frac{1}{2} & \frac{1}{6} & \frac{5}{6} & \frac{1}{6} \\ \frac{1}{6} & \frac{17}{18} & -\frac{5}{18} & -\frac{1}{18} \\ \frac{5}{6} & -\frac{5}{18} & -\frac{7}{18} & -\frac{5}{18} \\ \frac{1}{6} & -\frac{1}{18} & -\frac{5}{18} & \frac{17}{18} \end{pmatrix}$$

Oserviamo che le matrici sono simmetriche (potremmo omettere qualche calcolo), per verificare i risultati ottenuti possiamo verificare che $P \cdot x = \alpha e_1$, quindi che

$$\begin{pmatrix} \frac{1}{2} & \frac{1}{6} & \frac{5}{6} & \frac{1}{6} \\ \frac{1}{6} & \frac{17}{18} & -\frac{5}{18} & -\frac{1}{18} \\ \frac{5}{6} & -\frac{5}{18} & -\frac{7}{18} & -\frac{5}{18} \\ \frac{1}{6} & -\frac{1}{18} & -\frac{5}{18} & \frac{17}{18} \end{pmatrix} \begin{pmatrix} -3 \\ 1 \\ 5 \\ 1 \end{pmatrix} = \begin{pmatrix} \vdots \\ \vdots \\ \vdots \\ \vdots \end{pmatrix} \stackrel{?}{=} \begin{pmatrix} 6 \\ 0 \\ 0 \\ 0 \end{pmatrix}$$

Oppure possiamo verificare che $P^2 = I$, quindi che una generica colonna x abbia $\|x\|_2 = 1$ e che il prodotto scalare tra le colonne sia nullo.

30 Fattorizzazione QR

La matrice di Householder e' una matrice ortogonale Consideriamo nuvoamente una matrice generica

$$\begin{pmatrix} \times & \times & \times & \times \\ \times & \times & \times & \times \\ \times & \times & \times & \times \\ \times & \times & \times & \times \\ \times & \times & \times & \times \\ \times & \times & \times & \times \end{pmatrix}$$

Dal momento che vogliamo azzerare i valori evidenziati dobbiamo trovare:

$$\bullet P_1 \text{ s.t. } P_1 \cdot \begin{pmatrix} x \\ \times \\ \times \\ \times \\ \times \\ \times \end{pmatrix} = \begin{pmatrix} \alpha \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}$$

$$\bullet P_2 \text{ s.t. } P_2 \cdot \begin{pmatrix} x \\ \times \\ \times \\ \times \\ \times \\ \times \end{pmatrix} = \begin{pmatrix} \alpha \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}$$

$$\bullet P_3 \text{ s.t. } P_3 \cdot \begin{pmatrix} x \\ \times \\ \times \\ \times \\ \times \\ \times \end{pmatrix} = \begin{pmatrix} \alpha \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}$$

$$\bullet P_4 \text{ s.t. } P_4 \cdot \begin{pmatrix} x \\ \times \\ \times \\ \times \\ \times \\ \times \end{pmatrix} = \begin{pmatrix} \alpha \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}$$

Dal momento che pero' dobbiamo moltiplicare per delle matrici 6×6 , quindi orliamo P_2, P_3, P_4 con le entrate corrispondenti della matrice identica. Scriviamo quindi:

$$\underbrace{\begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix}}_{P_4} \underbrace{\begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix}}_{P_3} \underbrace{\begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix}}_{P_2} \cdot P_1 \cdot \begin{pmatrix} x & x & x & x \\ x & x & x & x \\ x & x & x & x \\ x & x & x & x \\ x & x & x & x \\ x & x & x & x \end{pmatrix}$$

Il risultato ottenuto, come precedentemente e' una matrice 4×6 formata da una matrice triangolare superiore R e due righe nulle. $\tilde{P}_4, \tilde{P}_3, \tilde{P}_2$ sono matrici ortogonali, quindi $\tilde{p}^t = \tilde{p}^{-1} := Q$, possiamo quindi scrivere $A = Q \begin{pmatrix} R \\ 0 \end{pmatrix}$

Riassumiamo

Algoritmo	Costo	Stabilita'
Riduzione di Gauss	$\frac{1}{2}n^2(m - \frac{n}{3})$	Solo con pivoting parziale
QR con Givens	$2n^2(m - \frac{n}{3})$	Sempre ^a
QR con Householder	$n^2(m - \frac{n}{3})$	Sempre

^aQuesto algoritmo e' particolarmente efficiente se opera su una **matrice sparsa** (n. elementi $\neq 0 \ll m \cdot n$), inoltre e' il piu' adatto alla parallelizzazione