



Calcolo numerico - In questi riassunti ho cercato di riassumere le informazioni che ritenevo più

Calcolo numerico (Politecnico di Torino)



Scan to open on Studocu

CALCOLO NUMERICO [MATLAB]

RAPPRESENTAZIONE FLOATING POINT.

$$a = (-1)^s p N^q$$

p = mantissa
 q = esponente
 N = base sistema numerazione

$\left. \begin{array}{l} p \text{ e } q \text{ è normalizzata} \\ \text{prima cifra di } p \text{ dopo "," deve essere } \neq 0 \end{array} \right\} \Rightarrow N^{-1} \leq |p| < 1$

NUMERI DI MACCHINA

= numeri con mantissa ed esponente rappresentabili in spazi a loro riservati in calcolatore \rightarrow numero finito

t = massimo numero cifre mantissa rappresentabili

$$m \leq q \leq M \quad m < 0, M > 0$$

- \rightarrow UNDERFLOW: $q < m$ \rightarrow approssimo a 0
- \rightarrow OVERFLOW: $q > M$ \rightarrow processo di arresto

$\left\{ \begin{array}{l} \text{Troncamento} \textcircled{1} \\ \text{ARROTONDAMENTO rounding to even} \textcircled{2} \end{array} \right. \rightarrow p \text{ approssimata a } \tilde{p} \text{ più vicina di macchina}$

\hookrightarrow per calcolo $\tilde{p} = p + \frac{1}{2} N^{-t}$

ERRORE ASSOLUTO: $|a - \tilde{a}| = e_a \quad \tilde{a} = \text{di macchina}$

RELATIVO: $\frac{|a - \tilde{a}|}{|a|} = e_r$

ERRORE DI ARROTONDAMENTO:

$$\textcircled{1} \quad |p - \tilde{p}| < N^{-t}$$

$$e_a < N^{-t}$$

$$e_r < N^{-t-1}$$

$$\epsilon_{ps} = N^{-t-1} = \epsilon$$

\hookrightarrow Più piccolo numero rappresentabile

$$\textcircled{2} \quad |p - \tilde{p}| \leq \frac{1}{2} N^{-t}$$

$$e_a \leq \frac{1}{2} N^{-t}$$

$$e_r \leq \frac{1}{2} N^{-t-1}$$

$$\text{precisione macchina} = \epsilon_m = \frac{1}{2} N^{-t-1}$$

IEEE 754, doppia precisione.
 $N=2$ 64 bit
 1 bit segno p
 52 bit = cifre p
 11 bit = q

ϵ_{ps}
 $\text{real min} = \text{più piccolo n° mac}$
 $\text{real max} = \text{più grande n° mac}$

OPERAZIONI DI MACCHINA $\oplus, \ominus, \otimes, \oslash$

Associa a due numeri di macchina un terzo numero di macchina, ottenuto arrotondando esatto risultato operazione

$$\bar{a}_1 \oplus \bar{a}_2 = \overline{a_1 + a_2} = (a_1 + a_2)(1 + e_{r\oplus})$$

i = con altri segni

\rightarrow Vale proprietà commutativa
 non valgono associativa e distributiva

Il calcolatore considera due espressioni equivalenti se: $\frac{|\bar{a}_1 - \bar{a}_2|}{|\bar{a}_1|} \approx \epsilon$ oppure $\frac{|\bar{a}_1 - \bar{a}_2|}{|\bar{a}_2|} \approx \epsilon$

CANCELLAZIONE NUMERICA

- Perdite cifre mantissa quando sottraggo due numeri quasi uguali arrotondati a numero di macchina
- Arrotondamento propaga errore
- \rightarrow Si può evitare conc. numerica \leq Razionalizzando Formule trigonometriche Polinomio Taylor

CONDIZIONAMENTO PROBLEMA

- Problema numerico: relazione non ambigua tra input (x) e output (y)
- Ben condizionato: $\frac{\|x - \tilde{x}\|}{\|x\|} = \frac{\|f(x) - f(\tilde{x})\|}{\|f(x)\|}$ \rightarrow perturbazione dei dati \rightarrow ingegneri troppo creativi

→ Studio condizion. verso numero di condiz. $K(f, x) \rightarrow \frac{|f(x) - \tilde{f}(x)|}{|f(x)|} \leq K(f, x) \frac{|x - \tilde{x}|}{|x|}$
 $\hookrightarrow \kappa \geq 1 \rightarrow$ Ben condizionato $\left\{ \begin{array}{l} \kappa = \text{cond} \log(A) \end{array} \right.$

STABILITÀ ALGORITMO

→ sequenza finita di operazioni

Se errore relativo associato a \tilde{y} ha stesso ordine grandezza di precisione di macchine o minore → operazioni non amplificano troppo errori indotti

APPROSSIMAZIONE

Sostituire f con \tilde{f} che imita comportamento ma è più facile operare

INTERPOLAZIONE POLINOMIALE (METODO INTERPOLAZIONE)

Scegliere una funzione e approssimarla a un polinomio

Assegnati $m+1$ punti esiste un solo $p_m(x)$ (polinomio interpolazione)
 $\hookrightarrow x_i =$ nodi interpolaz.

$$p_m(x) = c_2 x^m + c_2 x^{m-1} + \dots + c_m x + c_{m+1}$$

$c = \text{polyfit}(x, y, m) \rightarrow$ calcola e memorizza in c i coeff. di rappre. monomiale
 \hookrightarrow Length del polinomio di grado m interpolante x_i, y_i

$p = \text{polyval}(c, z) \rightarrow$ calcola e memorizza in p i valori che qualsiasi polinomio i cui coeff. sono in c , assume in z_i

RAPPRESENTAZIONE LAGRANGE (METODO INTERPOLAZIONE)

$$l_j(x_i) = \begin{cases} 1 & \text{se } i=j \\ 0 & \text{se } i \neq j \end{cases} \quad l_j(x) = \frac{(x-x_1) \dots (x-x_{j-1})(x-x_{j+1}) \dots (x-x_{m+1})}{(x_j-x_1) \dots (x_j-x_{j-1})(x_j-x_{j+1}) \dots (x_j-x_{m+1})} \quad \left\{ \begin{array}{l} \text{Polinomi} \\ \text{fundam.} \\ \text{Lagrange} \end{array} \right.$$

Interpolazione
 i dati: $(x_1, 0), \dots, (x_{j-1}, 0), (x_j, 1), (x_{j+1}, 0), \dots, (x_{m+1}, 0)$

$$p_m(x) = \sum_{j=1}^{m+1} y_j l_j(x)$$

ERRORE INTERPOLAZIONE: $\bar{E}_m(x) = f(x) - p_m(x)$

Anche se, $\forall x = x_i, \bar{E}_m(x_i) = 0$ non è detto che per $m \rightarrow \infty$ $\bar{E}_m(x) \rightarrow 0$

La convergenza uniforme garantita con nodi (Chebyshev: $z_i = -\cos\left(\frac{(i-1)\pi}{m}\right) \in [-1, 1]$)

FUNZIONI POLINOMIALI A TRATTI (Spline) (METODO INTERPOLAZIONE)

Si definisce funz. polin. a tratti di grado d associata a $[a, b]$ una f continua in $[a, b]$ e definita da unione tratti congiunti

Sono continue ma generalmente non derivabili in punti tra tratti
 Viste quando polinomio oscilla

Dato $a \equiv x_1 < x_2 < \dots < x_{m+1} \equiv b$, spline ordine d interpolante $f \equiv S_d(x)$ se:

- $S_d(x)$ è polinomio grado (al più) d $x \in [x_i, x_{i+1}]$ $i = 1, \dots, m$
- $S_d^{(k)}(x)$ di ordine k è continua in $[a, b]$ per $k = 0, 1, \dots, d-1$
- $S_d(x_i) = y_i$ $i = 1, \dots, m+1$

$\left\{ \begin{array}{l} S = \text{interp1}(x, y, z) \rightarrow \text{calcola e memorizza in } S \text{ valori che spline} \\ \text{lineare interpolante } x_i, y_i \text{ assume in } z \end{array} \right.$

Si definisce spline cubica una $f S_3(x)$ che:

- POLINOMIO GRADO 3 IN TUTTI I TRATTI: $S_3(x) = a_i + b_i x + c_i x^2 + d_i x^3$ $x \in [x_i, x_{i+1}]$

• REGOLARE NEI PUNTI DI RACCORDO: ($\in C^2([x_1, x_n])$)

$$S_3^{(k)} = S_3^{(k)}(x_1) \quad k=0, 1, 2$$

$$S_3(x_i) = y_i \quad i=1, \dots, m+1$$

$\left\{ \begin{array}{l} \text{Spline} \\ \text{not-a-knot} \rightarrow S = \text{spline}(x, y, z) \rightarrow \text{calcola e memorizza in } S \text{ valori due} \\ \text{spline cubica interpolante } x_i, y_i \text{ assieme} \\ \text{in } z \\ \text{Vincolata} \rightarrow S = \text{spline}(x, [y_i, y, y_m], z) \end{array} \right.$

RAPPRESENTAZIONE DI NEWTON (METODO DI INTERPOLAZIONE)

$$p_m(x) = f(x_1) + f[x_1, x_2](x-x_1) + f[x_1, x_2, x_3](x-x_1)(x-x_2) + \dots + f[x_1, x_n](x-x_1)(x-x_2)\dots(x-x_{n-1})$$

$$f[x_1, x_2] = \frac{f(x_2) - f(x_1)}{x_2 - x_1} \quad f[x_1, x_2, x_3] = \frac{f[x_2, x_3] - f[x_1, x_2]}{x_3 - x_1}$$

$$f[x_1, x_2, \dots, x_{m+1}] = \frac{f[x_2, x_3, \dots, x_{m+1}] - f[x_1, x_2, \dots, x_m]}{x_{m+1} - x_1}$$

SISTEMI LINEARI

NORME:

$$\|X\|_1 = \sum_{i=1}^m |x_i|$$

$$\|X\|_2 = \sqrt{\sum_{i=1}^m x_i^2} = \sqrt{X^T X} \quad \text{norma euclidea}$$

$$\|X\|_\infty = \max_{1 \leq i \leq m} |x_i|$$

$\left\{ \begin{array}{l} \text{norm}(X, 1) \\ \text{norm}(X, 2) \\ \text{norm}(X, \infty) \end{array} \right.$
 $\left\{ \begin{array}{l} \text{norm}(A, 1) \\ \text{norm}(A, 2) \\ \text{norm}(A, \infty) \end{array} \right.$

$$\|AX\| \leq \|A\| \|X\|$$

- $\|X\| \geq 0 \quad \forall X \in \mathbb{R}^m \quad \|X\| = 0 \rightarrow X = 0$
- $\|aX\| = |a| \|X\| \quad \forall a \in \mathbb{R}, \forall X \in \mathbb{R}^m$
- $\|X+Y\| \leq \|X\| + \|Y\| \quad \forall X, Y \in \mathbb{R}^m$

CONDIZIONAMENTO SISTEMA LINEARE

$$Ax = b$$

numero condizionamento: $K(A) = \|A\| \|A^{-1}\|$

$\text{cond}(A, 1) \rightarrow \text{cond. in norma } 1$

$\text{cond}(A, \infty) \rightarrow \text{cond. in norma } \infty$

$\text{hill}(m) \rightarrow \text{genera matrice Hilbert}$

$\text{vander}(n) \rightarrow \text{genera matrice Vandermonde}$

$\rightarrow \text{risolve sistemi lineari } A \setminus b$

$= 1$, ben condizionato
 (piccoli cambiamenti su
 dati = piccoli incrementi)
 $\gg 1$ mal condiz.

METODI NUMERICI

SISTEMI DIAGONALI: $x_i = \frac{b_i}{a_{ii}} \quad i=1, \dots, m$

Metodo sostituzione all'indietro: $\frac{m^2}{2}$

$$\left\{ \begin{array}{l} x_m = \frac{b_m}{a_{mm}} \\ x_i = \frac{b_i - \sum_{j=i+1}^m a_{ij} x_j}{a_{ii}} \quad i=m-1, \dots, 1 \end{array} \right.$$

\rightarrow SISTEMI TRIANGOLARI SUPERIORI

function $x = \text{indietro}(A, b)$
 $n = \text{length}(b);$
 $x = \text{zeros}(n, 1);$
 $x(n) = b(n) / A(n, n);$
 $S = A(i, i+1:n) * x(i+1:n);$
 $x(i) = (b(i) - S) / A(i, i);$
 end

Metodo di sostituzione in avanti: $\frac{n^2}{2}$

$$x_1 = \frac{b_1}{a_{11}}$$

$$x_i = \frac{b_i - \sum_{j=1}^{i-1} a_{ij} x_j}{a_{ii}} \quad i=2, \dots, n$$

```
function x = avanti(A, b)
m = length(b);
x = zeros(m, 1);
x(1) = b(1)/A(1,1);
for i = 2:m
    S = A(i, 1:i-1) * x(1:i-1);
    x(i) = (b(i) - S) / A(i, i);
end
```

Metodo eliminazioni Gauss: $\frac{n^3}{3}$

Da $Ax=b$ a $Ux=b$ U = matrice triang. sup.

$Ux=b \rightarrow$ mediante sostituz. indietro

Soluzione non cambia se: - si scambiano due equaz. del sistema
- si sostituisce a " " " una
combinazione lineare di eq. plane

esempio:

$$A = \begin{pmatrix} 2 & -1 & 1 & -2 \\ 0 & 2 & 0 & -1 \\ 1 & 0 & -2 & 1 \\ 0 & 2 & 1 & 1 \end{pmatrix} \quad b = \begin{pmatrix} 0 \\ 1 \\ 0 \\ 4 \end{pmatrix}$$

$k=1$ moltiplicatore $a_{ik} = \frac{a_{ik}}{a_{kk}}$

$a_{i1} \neq 0 \rightarrow 0.5$

Forse lo cambia riga

$a_{ij} = a_{ij} - a_{ik} a_{kj}$

$b_i = b_i - a_{ik} b_{kk}$

$$\rightarrow A = \begin{pmatrix} 2 & -1 & 1 & -2 \\ 0 & 2 & 0 & -1 \\ 1/2 & 1/2 & -3/2 & 2 \\ 0 & 2 & 1 & 1 \end{pmatrix} \quad b = \begin{pmatrix} 0 \\ 1 \\ 0 \\ 4 \end{pmatrix}$$

$k=...$

function x = gauss_nonscambi(A, b)

m = length(b)

for k = 1:m-1

for i = k+1:m

A(i, k) = A(i, k) / A(k, k);

for j = k+1:m

A(i, j) = A(i, j) - A(i, k) * A(k, j);

end

b(i) = b(i) - A(i, k) * b(k);

end

end

x = zeros(m, 1)

x(m) = b(m) / A(m, m);

for i = m-1:-2:1

x(i) = (b(i) - A(i, i+1:m) * x(i+1:m)) / A(i, i);

end

PIVOTING PARZIALE: Scegliere un a_{ik} che sia il più grande possibile in modo da avere scambio equazione, rende più stabile Gauss

Non serve se: A è diag-dominante per righe.
" " simmetrica e definita positiva

$$\begin{cases} x = A \backslash b \\ [L, U, P] = \text{lu}(A) \\ \hookrightarrow PA = LU \end{cases}$$

FATTORIZZAZIONE $PA=LU$

$$\text{Se } Ax=b \rightarrow PAx=Pb \rightarrow LUx=Pb$$

$$\begin{cases} LY = Pb \rightarrow y \\ UX = y \rightarrow x \end{cases} \quad \begin{cases} y = L \backslash Pb \\ x = L \backslash U \backslash Pb \end{cases}$$

$$\det(A) = (-1)^S \prod_{i=1}^n U_{ii}$$

$$\text{inv}(A) = PA=LU \rightarrow (PA)^{-1} = (LU)^{-1} \rightarrow A^{-1}P^{-1} = U^{-1}L^{-1} \rightarrow A^{-1}P = U^{-1}L^{-1}P \quad \text{inv}(A)$$

FATTORIZZAZIONE CHOLESKY: $\frac{n^3}{6}$

$$A = LU = R^T R$$

R = triang. sup
 $R^T =$ " inf

$$\text{chol}(A) = R$$

Calcolo inversa simmetrica positiva: $A^{-1} = R^{-1} (R^{-1})^T$ $\frac{2n^3}{3}$

FATTORIZZAZIONE QR

$$A = QR \quad \begin{cases} Q \in \mathbb{R}^{m,m} \text{ ortogonale} \\ R \in \mathbb{R}^{m,n} \rightarrow r_{ij} = 0 \quad i > j \end{cases} \quad [Q, R] = qr(A)$$

→ Riduzione sistema lineare determinato: $\frac{3}{2}n^2$

$$A^{m,n} Ax = b \rightarrow QRx = b \rightarrow Rx = Q^T b$$

→ Riduzione sistema lineare sovradeterminato:

Se non ammette soluzioni si cercano vettori x per cui il vettore residuo $Ax - b$ è piccolo
 $\hookrightarrow \|r\| = \|Ax - b\|$

$x \in X$ se $A^T A x = A^T b$
 Si riduce a un solo elemento x^* se
 ind. rango max

soluzione di
 min. norma
 $\|x^*\|_2 = \min_{x \in X} \|x\|_2$

AUTOVALORI

$Ax = \lambda x$ → autovettore di A $\{Ax = xD$

MATRICE
 DIAGONALE

λ è autovaleore se e solo se $A - \lambda I$ è singolare

$\{ \text{eig}(A) \rightarrow \text{vettore con tutti autovaleori}$

$$\text{QUOZIENTE DI RAYLEIGH: } r_A(x) = \frac{x^T A x}{x^T x}$$

$$[x, D] = \text{eig}(A)$$

autovet → autovaleori

↳ dato autovettore x , autovaleore λ associato
 è quoz. Rayleigh

A è simmetrica se e solo se λ sono tutti positivi

RAGGIO SPETTRALE: modulo di max autovaleori $\rightarrow \{ \max(|\text{abs}(\text{eig}(A))|) \}$

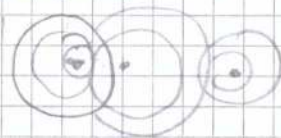
↳ norma 2 o SPETTRALE: $\|A\|_2 = \sqrt{\rho(A^T A)}$ $\{ \text{norm}(A, 2) \}$

Per trovare autovaleori di A nel piano complesso:

CERCHIO RIGA DI GERSHGORIN: $C_i(r) = \{ z \in \mathbb{C} : |z - a_{ii}| \leq \sum_{j=1, j \neq i}^n |a_{ij}| \}$ $\{ i=1, \dots, n \}$
 $C_i(c) = \{ z \in \mathbb{C} : |z - a_{ii}| \leq \sum_{j=1, j \neq i}^n |a_{ji}| \}$ $\{ i=1, \dots, n \}$

• Definiscono una regione di piano in cui ci sono tutti autovaleori di matrice

esempio:



vedi slides

↳ autovaleori cadono in intersezione dei cerchi riga e colonne

↳ Centro = elemento diagonale

Raggio cerchi = somma elementi su riga o colonna

PERTURBAZIONI: una perturbazione di 0,1% su elementi matrice corrisponde a variazione del 30% su autovaleori

TEOREMA BAUKER-FIKE:

A diagonalizz. e simmetr. $S^{-1}AS = D$

\tilde{A} = perturbazione di A e $\tilde{\lambda}$ autovaleore di $\tilde{A} \rightarrow \min_{1 \leq i \leq m} |\tilde{\lambda} - \lambda_i| \leq (K(S)) \|A - \tilde{A}\|$

$$= \|S\| \|S^{-1}\| \dots$$

Se A è simmetrica e diagonalizz. → calcolo autovaleori è ben condizionato

↳ Anche matrice Hilbert è ben condizionata per il calcolo degli autovaleori → no per traslazione

$\{ C = \text{cond}(\text{eig}(A)) \}$

METODI NUMERICI

2) Approssimazione di un solo autovettore

METODO DELLE POTENZE:

Calcola $\max \lambda$ e vettore spettrale $p(A)$
 Supponiamo A diagonalizzabile e un solo λ_{\max}
 x_1, \dots, x_n autovettori associati a $\lambda_1, \dots, \lambda_n$ autovalori
 $z = \alpha_1 x_1 + \alpha_2 x_2 + \dots + \alpha_n x_n \rightarrow Az = \alpha_1 Ax_1 + \dots + \alpha_n Ax_n = \alpha_1 \lambda_1 x_1 + \dots + \alpha_n \lambda_n x_n$
 $A^2 z = \alpha_1 \lambda_1^2 x_1 + \dots + \alpha_n \lambda_n^2 x_n \rightarrow A^m z = \alpha_1 \lambda_1^m x_1 + \dots + \alpha_n \lambda_n^m x_n$
 $z^m = \lambda_1^m \left(\alpha_1 x_1 + \alpha_2 \left(\frac{\lambda_2}{\lambda_1} \right)^m x_2 + \dots + \alpha_n \left(\frac{\lambda_n}{\lambda_1} \right)^m x_n \right) = \lambda_1^m y^m$
 $w^{(m)} = \frac{z^m}{\|z^m\|_2} \quad z^{m+1} = A w^{(m)} \quad \lambda_1^m = \frac{(w^{(m)})^T A w^{(m)}}{(w^{(m)})^T w^{(m)}} = (w^{(m)})^T A w^{(m)} = (w^{(m)})^T z^{m+1}$

```
function [lambda_max, w, m] = potenze(A, z, tol, m_max)
    w = z / norm(z)
    lambda_max = 0
    for m = 1:m_max
        z = A * w
        lambda_max = w' * z
        w = z / norm(z)
        if abs(lambda_max - lambda_max) <= tol * abs(lambda_max)
            break
        end
    end
    lambda_max = lambda_max;
end
```

La convergenza dipende da valore di primo λ_2
 non converge se $\lambda_2 = -\lambda_1$
 " " " " λ_1 e λ_2 sono complessi coniugati

METODO DELLE POTENZE INVERSE

$Ax = \lambda x \rightarrow (A - pI)x = Ax - px = (\lambda - p)x \rightarrow (A - pI)^{-1} x = \frac{1}{\lambda - p} x$
 $\lambda = p + \frac{1}{\mu}$ autovettore di max modulo $\frac{1}{\alpha - p}$

```
function [lambda_p, w, m] = potenze_inverse(A, p, z, tol, m_max)
    n = size(A, 1)
    w = z / norm(z)
    lambda_p = 0
    [L, U, P] = lu(A - p * eye(n));
    for m = 1:m_max
        y = L \ (P * w);
        z = U \ y;
        lambda_p = p + 1 / (w' * z);
        w = z / norm(z);
        if abs(lambda_p - lambda_p) <= tol * abs(lambda_p)
            break
        end
    end
    lambda_p = lambda_p;
end
```

autovettore di κ

$[X, D] = \text{eigs}(A, k)$
 (autovettori), κ

$[X, D] = \text{eigs}(A, k, p)$
 Costo una moltiplicazione più vicini a p

2) • Approssimazione di tutti gli autovalori:

METODO QR ~ CONVERGENZA è garantita se autovalori sono distinti

$$A_1 = A \rightarrow A_2 = Q_1 B_1$$

$$A_2 = B_1 Q_1 \rightarrow A_2 = Q_2 B_2 \dots A_2 = Q_2^T A_2 Q_2$$

Se A è simmetrica $\rightarrow A_\infty$ è diag.

Se A non " " , ma ha autoval. reali $\rightarrow A_\infty$ è Triang sup.

" " " " e ha alcuni autovalori complessi \rightarrow
 $\rightarrow A_\infty$ è quasi Triang sup.

function [V, m] = qr_base(A, tol, m_max)

for m = 2:m_max

[Q, R] = qr(A);

A = R * Q

if norm(tril(A, -1), 'inf') <= tol

break

end

d = diag(A)

~ Su diag. ha autovalori

Metodo costoso quando A è
densa, per ridurre costo:

$A \sim B$ - HESSEBERG SUP

(A NO SIMMETRICA)

TR, DIAGONALE

(A È SIMMETRICA)

B = HESS(A)

VALORI SINGOLARI e DECOMPOSIZIONE SVD

Consideriamo una matrice rettangolare:

Non si chiamano autovalori ma valori singolari!

$A \in \mathbb{R}^{m,n}$ $\exists U \in \mathbb{R}^{m,m}$ ortogonale, $V \in \mathbb{R}^{n,n}$ ortogonale

$\rightarrow U^T A V = S = \text{diag}(S_1, \dots, S_p) \in \mathbb{R}^{m,n} \rightarrow S_1 \geq S_2 \geq \dots \geq S_p \geq 0$ [valori
singolari]
 vettori colom. vettori colom.
sing. sinistri singolari destri
 $p = \min\{m, n\}$

$d = \text{svd}(A) \rightarrow$ restituisce vettore d contenente in ordine decrescente valori sing. A

$[U, S, V] = \text{svd}(A) \rightarrow A = U S V^T$ matrice diag. con elementi non negativi di
valori singolari

Calcolo valori singolari è un problema sempre ben condizionato

$A = U S V^T$ $S_1 \geq S_2 \geq \dots \geq S_r > S_{r+1} = S_p = 0$ $r < p$
 $r \neq 0$

• $r(A) = r$

• $A = U_r S_r V_r^T$ - $U_r \in \mathbb{R}^{m,r}$ matrice con colonne u_1, \dots, u_r
 $V_r \in \mathbb{R}^{n,r}$ " " " " v_1, \dots, v_r
 $S_r \in \mathbb{R}^{r,r}$ " " " " diag. con elementi S_1, \dots, S_r

• $k \leq r$ $A_k = \sum_{i=1}^k S_i u_i v_i^T$ e $B_k = \{B: r(B) = k\} \rightarrow \min_{B \in B_k} \|A - B\|_2 = \|A - A_k\|_2 = S_{k+1}$

Applicazioni SVD:

- rango matrice = n° valori singolari $\neq 0$ (MAGGIORI DI UNA
CERTA TOLLERANZA)

$\text{rank}(A) \rightarrow$ calcolo
con tolleranza
di default

- condizionamento: $\|A\|_2 = \sqrt{\rho(S^2)} = \lambda_L$

$$\kappa_2(A) = \|A\|_2 \|A^{-1}\|_2 = \frac{S_1}{S_m}$$

- risoluzione sistema lineare:

$$L \rightarrow U S V^T x = b \rightarrow S V^T x = U^T b \rightarrow \begin{cases} S y = U^T b \rightarrow y \\ V^T x = y \rightarrow x = V y \end{cases}$$

- risoluzione sistema sovradeterminato:

$$A \in \mathbb{R}^{m,n} \quad m > n \quad r(A) \leq n \quad \|A x - b\|_2^2 = \|S V x - U^T b\|_2^2 = \left\| \begin{pmatrix} S_0 \\ 0 \end{pmatrix} \begin{pmatrix} y_1 \\ y_2 \end{pmatrix} - \begin{pmatrix} c_1 \\ c_2 \end{pmatrix} \right\|_2^2$$

$$= \left\| \begin{pmatrix} \tilde{S}y_1 - c_2 \\ -c_2 \end{pmatrix} \right\|_2^2 = \|\tilde{S}y_1 - c_2\|_2^2 + \|c_2\|_2^2 = \min$$

$$(y_i^*) = \frac{U_i^T b}{S_i} \quad i =$$

$$x^* = V y^* = A^+ b \quad \left\{ \begin{array}{l} x = \text{plm}(A) \rightarrow A^+ \end{array} \right.$$

- risoluzione sistema non determinate

$$Ax = b \rightarrow SV^T x = U^T b \rightarrow \begin{cases} Sy = U^T b \\ V^T x = y \end{cases}$$

$$y_i^* = \frac{C_i}{S_i}$$

$$x^* = V y^*$$