Calcolo Scientifico

GGC

giuseppe.colabufo.2016@polytechnique.org

2020

Indice

1 Introduzione

Piccolo avviso: i teoremi con il simbolo $\diamondsuit$ sono stati dimostrati a lezione.

1.1 Numero di condizionamento

Definizione 1.1. Il numero condizionamento di $A \in \mathbb{C}^{n\times n}$ è dato da $\mu(A) := \|A\|\|A^{-1}\|$

Osservazione 1. Se A è unitaria, in $\|\cdot\|_2$ vale $\mu_2(A) = 1$.

Osservazione 2. Se A è diagonalizzabile: $A = SDS^{-1}$ e vale $T^{-1}AT = B$ Hessenberg superiore, per Bauer-Fike, l'errore nel calcolo degli autovalori dipende da

$$\mu(T^{-1}S) = \|T^{-1}S\|\|S^{-1}T\| \leq \mu(T)\mu(S)$$

quindi per minimizzare il condizionamento devo minimizzare $\mu(T)$ (sperando ad esempio in T unitaria).

Oss. 3: Numero di condizionamento per matrici speciali.

- Se T è una matrice elementare di Gauss, $\|T\|_\infty \leq 2$ e quindi $\mu_\infty(T) \leq 4$.
- Per matrici T elementari di Householder o di Givens vale $\mu_2(T) = 1$ (sono ortogonali).

Definizione 1.2. Sia $A \in \mathbb{C}^{m\times n}$ di rango k. Si dice numero di condizionamento di A il valore $\mu(A) := \|A\|\|A^+\|$.

Osservazione 4. In norma 2 vale $\mu_2(A) = \frac{\sigma_1}{\sigma_k}$.

2 Premesse

Si usa praticamente sempre che:

1. Una matrice A si può riportare tramite trasformazioni di similitudine ad una matrice B in forma normale di Hessemberg superiore.
2. Se A è hermitiana, si può riportare in forma tridiagonale.

Infatti:

Oss. 1: Trasformazione 1. Supponiamo A hermitiana e consideriamo la seguente successione di matrici:

$$\begin{aligned} A^{(1)} &= A \\ A^{(k+1)} &= T_k^{-1} A^{(k)} T_k \quad k = 1, \dots, m-1 \\ A^{(m)} &= B \end{aligned}$$

dove $T_k = I - \beta_k u_k u_k^H$ soono matrici di Householder. Se $T = T_1 T_2 \cdots T_{m-1}$, $B = T^{-1} A T$ Passi nella costruzione:

1. $P^{(1)} \in \mathbb{C}^{(n-1)\times(n-1)}$ tale che $P^{(1)} a_1 = \alpha_1 e_1$ dove $e_1 \in \mathbb{R}^{n-1}$.
2. $P^{(1)} = I - \beta_1 v_1 v_1^H$ con $v_1 = a_1 + \operatorname{sgn}(a_{21}^{(1)}) \|a_1\|_2 e_1$ da cui si ricava α_1 .
3. $T_1 = diag(1, P^{(1)}) = I - \beta (0, v_1)^T (0, v_1^H)$
4. iterando, al k -esimo passo la matrice A è tridiagonale in $(k-1)\times(k-1)$, e si modifica alla riga e colonna k .
5. L'unica cosa da calcolare è il costo di $P^{(k)} B^{(k)} P^{(k)}$ che modifica l'ultima parte di A .
6. esplicitando i conti, ponendo $r_k = \beta_k B^{(k)} v_k$ e $q_k = r_k - \frac{1}{2}\beta_k (r_k^H v_k) v_k)$ si ha:
$$P^{(k)} B^{(k)} P^{(k)} = B^{(k)} - q_k v_k^H - (q_k v_k^H)^H$$
7. i costi dominanti sono per il calcolo di $q_k v_k^H$ che è $(n-k)^2$ moltiplicazioni e di $B^{(k)} v_k$ che ha lo stesso costo.
8. sommando su k , il costo totale è $\frac{2}{3} n^3$.

Se A non fosse hermitiana si otterrebbe un costo totale di $\frac{5}{3} n^3$.

Oss. 2: Remind sulla fattorizzazione QR.

- esiste anche per matrici rettangolari;
- Q è una matrice unitaria;

- R è triangolare superiore;
- l'unicità non è garantita se non a meno di matrici di fase;
- una matrice di fase è $S = \begin{pmatrix} \theta_1 & & \\ & \ddots & \\ & & \theta_n \end{pmatrix}$ con $|\theta_i| = 1$, diagonale e unitaria.

Oss. 3: Norme matriciali di matrici non quadrate. Data $A \in \mathbb{C}^{m\times n}$ per $m, n \in \mathbb{N}$ qualsiasi, si definisce

$$\begin{aligned} \|\cdot\| \colon \mathbb{C}^{m\times n} &\to \mathbb{R} \\ A &\mapsto \max_{\|x\|=1} \|Ax\| \end{aligned}$$

Si verifica che la funzione soddisfa le proprietà di norma e che $\|A\|_2 = \sqrt{\rho(A^H A)}$

Oss. 4: Disuguaglianza di Kantorovich. Se A definita positiva (con λ_1 e λ_n minimo e massimo autovalore), per ogni $x \neq 0$ si ha

$$\frac{(x^H x)^2}{(x^H Ax)(x^H A^{-1} x)} \geq 4\frac{\lambda_1 \lambda_n}{(\lambda_1 + \lambda_n)^2}$$

2.1 Matrix pencil

Definizione 2.1. Siano $A, B \in \mathbb{C}^{m\times n}$ e $\lambda \in \mathbb{C}$. La matrice $A - \lambda B$ si chiama ***matrix pencil*** (o semplicemente *pencil*).

Definizione 2.2. Siano $A, B \in \mathbb{C}^{m\times n}$ e $\lambda \in \mathbb{C}$. Se $\det(A - \lambda B)$ non è identicamente nullo, il pencil si dice *regolare*.

Se il pencil è regolare, $p(\lambda) := \det(A - \lambda B)$ è il polinomio caratteristico del pencil e le sue radici si dicono autovalori del pencil. Se $\deg p < n$, allora per definizione ∞ è un autovalore del pencil.
Il problema generalizzato agli autovalori è risolvere

$$Ax = \lambda Bx \quad x \neq 0$$

Oss. 5: Oss. 1. Se $\det B \neq 0$ allora $B^1 Ax = \lambda x$ e cioè gli autovalori generalizzati del pencil coincidono con gli autovalori di $B^{-1}A$. Ne segue che in questo caso ∞ non può essere un autovalore generalizzato.

Oss. 6: Oss. 2. Se $\det A \neq 0$ si ha $A^{-1}Bx = \frac{1}{\lambda}x$ e cioè gli autovalori generalizzati del pencil sono i reciproci degli autovalori di $A^{-1}B$. In questo caso se $A^{-1}B$ ha un autovalore 0 , ∞ è un autovalore generalizzato.

Esempio Se $\det B = 0$ e quindi $\mathrm{rk}B < n$ si possono presentare varie situazioni:

- $$A = \begin{pmatrix} 1 & 2 \\ 0 & 2 \end{pmatrix} \quad B = \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix}$$
  con $p(\lambda) = 2(1-\lambda)$ e quindi autovalori generalizzati λ, ∞ .
- $$A = \begin{pmatrix} 1 & 2 \\ 0 & 2 \end{pmatrix} \quad B = \begin{pmatrix} 1 & 0 \\ 1 & 0 \end{pmatrix}$$
  con $p(\lambda) = 2$ e quindi $\nexists \lambda \in \mathbb{C}\ p(\lambda) = 0$, ovvero l'unico autovalore generalizzato è ∞ .
- $$A = \begin{pmatrix} 1 & 2 \\ 0 & 0 \end{pmatrix} \quad B = \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix}$$
  con $p(\lambda) = 0$ e quindi l'insieme degli autovalori generalizzati è $\mathbb{C}$.

Se $\det B \neq 0$ per calcolare l'inversa:

1) risolvo n sistemi lineari $BC = A$;
2) trovo gli autovalori di C utilizzando il metodo QR

Tuttavia se B non è ben condizionata, il risultato non è affidabile.

Teorema 2.1. *Siano $A, B \in \mathbb{C}^{n\times n}$ con $A - \lambda B$ pencil regolare. Allora esistono Q_L, Q_R unitarie tali che $Q_L A Q_R = T_A$ e $Q_L B Q_R = T_B$ triangolari superiori e gli autovalori di $A - \lambda B$ sono $\frac{(T_A)_{ii}}{(T_B)_{ii}}$.*

♢

Siano $A = A^T$ e $B = B^T$ definita positiva. Allora $A - \lambda B$ è una **symmetric definite pencil** e se X è unitaria, $X^T A X - \lambda X^T B X$ è ancora simmetrica e definita positiva.

Teorema 2.2 (Martin - Wilkinson, 1968)**.** *Sia $A - \lambda B$ un symmetric definite pencil. Allora $\exists X$ invertibile tale che $X^T A X = diag(\alpha_1, \ldots, \alpha_n)$ e $X^T B X = diag(\beta_1, \ldots, \beta_n)$ e gli autovalori del pencil sono $\frac{\alpha_i}{\beta_i} \in \mathbb{R}$ (finiti).*

♢

Osservazione 7. Poiché calcolare l'inversa è computazionalmente costoso, si procede come segue:

$$F = AL^{-T} \iff FL^T = A$$
$$H = H^{-1}F \iff LH = F$$

si risolvono i sistemi lineari ed essendo H simmetrica è sufficiente calcolare gli elementi sotto la diagonale. Il costo totale è di $\frac{2}{3}n^3$ moltiplicazioni.

2.2 Il prodotto di Kronecker

Date due matrici $A \in \mathbb{C}^{m\times n}$ e $B \in \mathbb{C}^{p\times q}$ definiamo prodotto di Kroecker di A e B la matrice

$$A \otimes B = \begin{pmatrix} a_{11}B & \dots & a_{1n}B \\ \vdots & \ddots & \vdots \\ a_{m1}B & \dots & a_{mn}B \end{pmatrix} \in \mathbb{C}^{(mp)\times(nq)}$$

Le principali proprietà del prodotto di Kronecker sono:

1) $(A \otimes B)(C \otimes D) = (AC) \otimes (BD)$

2) $(A \otimes B)^T = A^T \otimes B^T$

2.3 Il Metodo di Newton

Riassumiamo in breve il metodo di Newton.

3 Matrici Speciali

Definizione 3.1. Una matrice H si dice elementare di Householder se esistono $\sigma \in \mathbb{R} \setminus \{0\}$ e $w \in \mathbb{C}^n$ tali che

$$H = I - \sigma w w^H \quad \sigma = \frac{2}{w^H w} = \frac{2}{\|w\|_2}$$

Oss. 1: Proprietà di Householder.

- *hermitiana* $H = H^H$
- *unitaria* $H^H H = H H^H = I$
- $\det H = -1$

Definizione 3.2. Una matrice $A \in \mathbb{C}^{m \times n}$ è detta *Cauchy-like* se i suoi coefficienti sono della forma

$$(A)_{ij} = a_{ij} = \frac{1}{x_i - y_j} \quad x_i - y_j \neq 0 \quad i = 1, \ldots, m \;\; j = 1, \ldots, m$$

dove $x_i \in \mathbb{C}$ sono tutti distinti e $y_j \in \mathbb{C}$ altrettanto.

Oss. 2: Obs.. Una *matrice di Hilbert* è Cauchy-like con $x_i - y_j = i + j - 1$.

4 Stima di autovalori

Teorema 4.1 (Bauer Fike)**.** *Sia $A \in \mathbb{C}^{n\times n}$ diagonalizzabile tramite T. Sia $\delta A \in \mathbb{C}^{n\times n}$ e sia ξ autovalore per $A+\delta A$. Allora $\exists \lambda$ autovalore di A tale che*

$$|\lambda - \xi| \leq \mu(T)\|\delta A\|$$

Teorema 4.2. *Sia $A \in \mathbb{C}^{n\times n}$ e sia λ un suo autovalore di molteplicità algebrica 1. Siano $x, y \in \mathbb{C}^n$ rispettivamente un autovettore destro e sinistro relativi a λ. Sia $F \in \mathbb{C}^{n\times n}$ e $\lambda(\varepsilon)$ un autovalore per $A+\varepsilon F$.*
Allora vale $\lambda(\varepsilon) - \lambda \doteq \varepsilon \frac{y^H F x}{y^H x}$

$\diamondsuit$

Osservazione 1.

- Usando Cauchy-Swartz si ottiene $|\lambda(\varepsilon) - \lambda| \leq \frac{\|\varepsilon F\|_2}{|y^H x|}$
- Se A è normale, $y = x$ e si ritrova la maggiorazione di Bauer-Fike perché $\|x\|_2 = 1$.

Teorema 4.3. *Sia $A \in \mathbb{C}^{n\times n}$ e sia λ un suo autovalore di molteplicità algebrica > 1 e geometrica τ_λ. Se $C^{(1)}, \ldots, C^{(\tau_\lambda)}$ sono i blocchi di Jordan di ordine $\leq \xi$, allora detto $\lambda(\varepsilon)$ un autovalore della matrice perturbata $A+\varepsilon F$, si ha*

$$|\lambda(\varepsilon) - \lambda| \leq \gamma \varepsilon^{1/\xi} \quad \gamma > 0$$

4.1 Usare Newton senza conoscere i polinomi

Oss. 2: Oss. 1. Per una matrice tridiagonale $(\bar{\beta}_i, \alpha_i, \beta_i)$ non è riduttivo suppore che i $\beta_i \neq 0$, infatti potrei altrimenti spezzare la matrice in due o più tridiagonali in cui l'assunzione è vera. In questo modo la matrice è irriducibile. Si ha che

$$\begin{aligned} &P_0(\lambda) = 1 \\ &P_1(\lambda) = \alpha_1 - \lambda \\ &P_i(\lambda) = \det(B_i - \lambda I) = (\alpha_i - \lambda)P_{i-1}(\lambda) - |\beta_i|^2 P_{i-2}(\lambda) \quad \text{per } i = 2, \ldots, n. \end{aligned}$$

In questo modo, senza calcolare i coefficienti del polinomio caratteristico posso calcolare il suo valore in un punto e utilizzare il metodo di Newton

per una approssimazione degli autovalori. Vale un analogo per le derivate:

$$\begin{aligned} P_0'(\lambda) &= 0 \\ P_1'(\lambda) &= -1 \\ P_i'(\lambda) &= -P_{i-1}'(\lambda) + (\alpha_i - \lambda)P_{i-1}'(\lambda) - |\beta_i|^2 P_{i-2}(\lambda) \end{aligned}$$

Notiamo che il costo fin qui è $O(n)$. Ora Newton costruisce la successione:

$$\lambda_{k+1} = \lambda_k - \frac{P_n(\lambda)}{P_n'(\lambda)}$$

il cui costo per passo è ancora $O(n)$. Per applicare questo metodo bisogna aver già localizzato in qualche modo l'autovalore.

Oss. 3: Prop. I. Se ξ è autovalore di B_n, allora ξ **non** è autovalore di B_{n-1}.
Consideriamo

$$B_n = \left(\begin{array}{cc|c} & & 0 \\ & B_{n-1} & \beta_n \\ \hline 0 & \beta_n & \alpha_n \end{array}\right)$$

con B_{n-1} tridiagonale simmetrica (reale). Allora B_{n-1} si può diagonalizzare tramite matrice ortogonale Q_{n-1}. Se $\hat{Q}$ =diag$(Q_{n-1}, 1)$, allora

$$\hat{Q}B_n\hat{Q}^T = \begin{pmatrix} D_{n-1} & w \\ w^T & \alpha_n \end{pmatrix} =: F_n$$

è una matrice *a freccia*.

$$F_n = \begin{pmatrix} a & & c \\ & & \\ d & & b \end{pmatrix}$$

[thick]ab [thick]cb [thick]db

Osservazione 4. Gli autovalori si conservano per trasformazioni ortogonali: $\det(F_n - \lambda I) = \det(B_n - \lambda I)$.

Lemma 4.4. *Se* $M = \begin{pmatrix} A & B \\ C & D \end{pmatrix}$ *con* A, D *quadrate e* $\det A \neq 0$, *allora detta* $\Gamma = D - CA^{-1}B$ *il complemento di Schur di* A *nella matrice* M, *vale:*

$$M = \begin{pmatrix} I & 0 \\ CA^{-1} & I \end{pmatrix} \begin{pmatrix} A & 0 \\ 0 & \Gamma \end{pmatrix} \begin{pmatrix} I & A^{-1}B \\ 0 & I \end{pmatrix}$$

Poiché vale

$$F_n-\lambda I = \begin{pmatrix} E_{n-1} & w \\ w^T & \alpha_n-\lambda \end{pmatrix} = \left(\begin{array}{ccc|c} \lambda_1^{(n-1)}-\lambda & & 0 & \\ & \ddots & & w \\ 0 & & \lambda_{n-1}^{(n-1)}-\lambda & \\ \hline & w^T & & \alpha_n-\lambda \end{array}\right)$$

e per la **Prop. I** possiamo applicare il lemma (semplice conto), segue che

$$\begin{aligned} P_n(\lambda) &= \prod_{j=1}^{n-1}(\lambda_j^{(n-1)}-\lambda)\cdot\left[(\alpha_n-\lambda)-\sum_{i=1}^{n-1}\frac{w_i^2}{\lambda_i^{(n-1)}-\lambda}\right] \\ &=(\alpha_n-\lambda)\prod_{j=1}^{n-1}(\lambda_j^{(n-1)}-\lambda)-\sum_{i=1}^{n-1}w_i^2\prod_{j=1,j\neq i}^{n-1}(\lambda_j^{(n-1)}-\lambda) \end{aligned}$$

e se ξ è autovalore dovendo essere $P_n(\xi)=0$, segue che

$$g(\xi) := \alpha_n-\xi-\sum_{i=1}^{n-1}\frac{w_i^2}{\lambda_i^{(n-1)}-\xi}$$

è tale che $g'(\xi)<0$ e ha degli asintoti verticali in corrispondenza di $\lambda_j^{(n-1)}$ il che porta alla

Oss. 5: Prop. II. Gli autovalori di B_n sono "separati" dagli autovalori di B_{n-1}.

Definizione 4.1. Una successione di polinomi $P_i(\lambda)$ che verifica:

i) $P_0(\lambda)$ non cambia segno;

ii) $P_i(\lambda)=0 \Rightarrow P_{i-1}(\lambda)P_{i+i}(\lambda)<0$ per $i=1,\dots,n-1$;

iii) $P_n(\lambda)=0 \Rightarrow P_n'(\lambda)P_{n-1}(\lambda)<0$

si dice **successione di Sturm**

Teorema 4.5. *Se B è tridiagonale ed hermitiana, con $\beta_i\neq 0$, la successione di polinomi sopra definita è una successione di Sturm.*

$\diamondsuit$

Fissato λ^* considero la successione $P_i(\lambda^*)$. Detto $w(\lambda^*)$ il numero di variazioni di segno nella successione si ottiene il seguente teorema (se $P_i(\lambda^*)=0$ per convenzione il suo segno è quello di $P_{i-1}(\lambda^*)$):

Teorema 4.6. *Se $\{P_i(\lambda)\}_{i=0}^n$ è una successione di Sturm, il numero $w(b)-w(a)$ è il numero di zeri di $P_n(\lambda)$ nell'intervallo $[a,b)$.*

$\diamondsuit$

Consideriamo ora gli autovalori di B_n: $\lambda_1 > \lambda_2 > \cdots > \lambda_n$. Se con i teoremi di Gershgorin o Hirsh individuo un intervallo $[a_0, b_0)$ in cui cade λ_k, posso migliorare la localizzazione come segue:

- sia $\xi = \frac{1}{2}(a_0 + b_0)$;
- se $w(\xi) \geq n - k + 1$ allora $\lambda_k \in [a_0, \xi)$
- se $w(\xi) < n - k + 1$ allora $\lambda_k \in [\xi, b_0)$
- itero il ragionamento finché necessario.

4.2 Metodo di Hyman

Oss. 6: Info a caso.

- Dal 1957;
- si applica a matrici quadrate complesse irriducibili in forma di Hessenberg superiore;

Fissato λ si determinano un vettore $x \in \mathbb{C}^n$ tale che $x_n = 1$ e uno scalare γ per cui valga:

$$(A - \lambda I)x = \gamma e_1$$

$$\begin{pmatrix} a_{11}-\lambda & a_{12} & & \dots & a_{1n} \\ a_{21} & a_{22}-\lambda & & & \\ \vdots & & \ddots & \ddots & \\ 0 & & \dots & a_{nn-1} & a_{nn} \end{pmatrix} \begin{pmatrix} x_1 \\ \vdots \\ x_{n-1} \\ 1 \end{pmatrix} = \begin{pmatrix} \gamma \\ 0 \\ \vdots \\ 0 \end{pmatrix}$$

a questo punto con un *back-solve* ricavo $x_{n-1}, \dots, x_1, \gamma$ pagando $\frac{n^2}{2}$ moltiplicazioni. Utilizzando la regola di Cramer si nota:

$$x_n = \frac{\det \begin{pmatrix} a_{11}-\lambda & \dots & \gamma \\ & \ddots & \vdots \\ & a_{nn-1} & 0 \end{pmatrix}}{\det(A - \lambda I)} = 1$$

da cui

$$P_n(\lambda) = (-1)^{n+1}\gamma a_{21}a_{32}\cdots a_{nn-1}$$

(e il costo fin qui è $O(n^2)$)

$$P_n'(\lambda) = \frac{d}{dt}(det(A - \lambda I)) = (-1)^{n+1}\gamma' a_{21}\cdots a_{nn-1}$$

e il calcolo di γ' si fa in costo $O(n^2)$ (compreso il calcolo di $x'(\lambda)$) con un *back-solve*:

$$\begin{aligned} (A - \lambda I)x(\lambda) &= \gamma(\lambda)e_1 \\ -x(\lambda) + (A - \lambda I)x'(\lambda) &= \gamma'(\lambda)e_1 \end{aligned}$$

A questo punto con il metodo di Newton trovo un'approssimazione della radice.

4.3 Metodo `DIVIDE ET IMPERA`

Oss. 7: Info a caso.

- Dal 1981 by Cuppen;
- si applica a matrici quadrate complesse tridiagonali;
- (quindi a matrici hermitiane ché si portano in questa forma);

Per semplicità consideriamo T =tridiag(b_i, a_i, b_i) reale.

$$T = \begin{pmatrix} a_1 & b_1 & & \\ b_1 & \ddots & \ddots & \\ & \ddots & \ddots & b_{n-1} \\ & & b_{n-1} & a_n \end{pmatrix}$$

$$= \left(\begin{array}{c|c} T_1 & \\ \hline & T_2 \end{array}\right) + \left(\begin{array}{cc|cc} & & & \mathbf{0} \\ & 0 & b_m & 0 \\ \hline & b_m & & \\ \mathbf{0} & & 0 & \end{array}\right)$$

dove le matrici $T_1 \in \mathbb{R}^{m\times m}$ e $T_2 \in \mathbb{R}^{n-m\times n-m}$ sono tridiagonali.
Posso modificare le posizioni (m, m) e $(m+1, m+1)$ in modo che la matrice di correzione abbia rango 1: sottraggo b_m da a_{mm} e a_{m+1m+1} e allora si riscrive

$$T = \left(\begin{array}{c|c} T_1 & \\ \hline & T_2 \end{array}\right) + b_m \begin{pmatrix} 0 \\ \vdots \\ 1 \\ 1 \\ \vdots \\ 0 \end{pmatrix} \begin{pmatrix} 0 & \cdots & 1 & 1 & \cdots & 0 \end{pmatrix}$$

$$= \left(\begin{array}{c|c} T_1 & \\ \hline & T_2 \end{array}\right) + b_m v v^T$$

Supponiamo ora di conoscere gli autovalori di T_1 e T_2 e i corrispondenti autovettori.
Allora $T_1 = Q_1\Gamma_1 Q_1^T$ e $T_2 = Q_2\Gamma_2 Q_2^T$, da cui se u è tale che

$$\begin{pmatrix} Q_1 & \\ & Q_2 \end{pmatrix} u = v$$

si ottiene

$$\begin{aligned} T &= \left(\begin{array}{c|c} Q_1\Gamma_1 Q_1^T & \\ \hline & Q_2\Gamma_2 Q_2^T \end{array}\right) + b_m vv^T \\ &= \left(\begin{array}{c|c} Q_1 & \\ \hline & Q_2 \end{array}\right) \left[\left(\begin{array}{c|c} \Gamma_1 & \\ \hline & \Gamma_2 \end{array}\right) + b_m uu^T\right] \left(\begin{array}{c|c} Q_1^T & \\ \hline & Q_2^T \end{array}\right) \end{aligned}$$

Chiamando $D = (D_{ij}) = \text{diag}(d_i)$ la matrice diagonale, e posto $\rho := b_m$ per conoscere gli autovalori di T serve calcolare

$$\begin{aligned} \det(D + \rho uu^T - \lambda I) &= \det((D - \lambda I)(I + \rho(D - \lambda I)^{-1} uu^T)) = 0 \\ &\iff \det(I + \rho(D - \lambda I)^{-1} uu^T) = 0 \\ &\iff \det(I + uu^T) = 1 + \rho \sum_{i=1}^{n} \frac{u_i^2}{\lambda_i - \lambda} = 0 \end{aligned}$$

Ovvero gli autovalori di T sono le radici di

$$f(\lambda) = 1 + \rho \sum_{i=1}^{n} \frac{u_i^2}{\lambda_i - \lambda}$$

di cui notiamo che la derivata prima

$$f'(\lambda) = \rho \sum_{i=1}^{n} \frac{u_i^2}{(\lambda_i - \lambda)^2}$$

ha lo stesso segno di ρ.

Oss. 8: Oss. 1. Il costo per calcolare gli zeri è $O(n)$ (precisamente $3n + 2$).

Oss. 9: Oss. 2. Conoscendo Q_1 e Q_2 il calcolo di u è gratis: basta mettere l'ultima colonna di Q_1^T sulla prima colonna di Q_2^T.

Proposizione 4.7. *Sia α un autovalore di $D + \rho uu^T$. Allora $(D - \alpha I)^{-1}u$ è autovettore relativo all'autovalore α.*

♢

Se Q' è la matrice che ha per colonne gli autovettori di $D + \rho uu^T$, gli autovettori di T sono in $\left(\begin{array}{c|c} Q_1^T & \\ \hline & Q_2^T \end{array}\right) Q'$. La matrice Q' è *Cauchy-like* perché i suoi elementi sono della forma $\frac{u_i}{d_i - \alpha}$

Osservazione 10. Il costo per gli autovettori di T sarebbe $O(n^3)$ ma il prodotto di matrici di cui una *Cauchy-like* si fa in tempo $O(n^2 \log n)$.

4.4 Metodo QR

Oss. 11: Info a caso.

- Dal 1961 by Francis;
- precedentemente si basava sulla fattorizzazione LU
- permette di calcolare tutti gli autovalori di una generica matrice complessa;

Costruiamo una successione di matrici $\{A_k\}$:

$$\begin{aligned} A_1 &= A = Q_1 R_1 \\ A_2 &= R_1 Q_1 = Q_2 R_2 \\ &\dots \\ A_k &= R_{k-1} Q_{k-1} = Q_k R_k \end{aligned}$$

ciclando fino a convergenza. Si nota che tutte le matrici sono simili mediante trasformazione unitarie e quindi il problema è ben condizionato:

$$A_{k+1} = R_k Q_k = (Q_k^H Q_k) R_k Q_k = Q_k^H A_k Q_k$$

Teorema 4.8. *Sia $A \in \mathbb{C}^{n\times n}$ con autovalori λ_i tali che $|\lambda_1| > |\lambda_2| > \dots > |\lambda_n|$. Supponiamo che A sia diagonalizzabile tramite X: $A = XDX^{-1}$ e che esista la fattorizzazione $X^{-1} = LU$. Allora $\exists S_k$ matrici di fase tali che*

$$\lim_{k\to+\infty} S_k^H R_k S_{k-1} = \lim_{k\to+\infty} S_{k-1}^H A_k S_k = T$$

dove T è triangolare superiore, e

$$\lim_{k\to+\infty} S_{k-1}^H Q_k S_k = I$$

$\diamondsuit$

Oss. 12: Costo computazionale. Per il calcolo di $Q_k R_k$ e quello di $R_k Q_k$ è un $O(n^3)$ e quindi un $O(n^3)$ ad ogni passo. Però si può fare di meglio: se tramite matrici di Householder mi riporto ad avere A in forma di Hessenberg superiore (o tridiagonale se era hermitiana) applicando il metodo QR ottengo un costo per passo dell'ordine di $O(n^2)$ (rispettivamente $O(n)$) e ad ogni passo il metodo conserva la struttura. C'è da dire però che la trasformazione costa già $O(n^3)$.

4.5 Criterio di arresto

Fissata una tolleranza ε si procede col metodo finché $a^{(k)}_{p+1p}$ è sufficientemente piccolo: $p \in [1, n)$ e $|a^{(k)}_{p+1p}| \leq \varepsilon(|a^{(k)}_{pp}| + |a^{(k)}_{p+1p+1}|)$,

$$\left(\begin{array}{c|c} \ddots & \\ & 0 \\ a^{(k)}_{pp} & \\ \hline a^{(k)}_{p+1p} & a^{(k)}_{p+1p+1} \\ 0 & \ddots \end{array}\right)$$

annullo il blocco in basso a sinistra e lavoro su due matrici più piccole. La velocità di convergenza (nelle ipotesi del teorems) dipende da $|\frac{\lambda_i}{\lambda_j}|$ e in particolare quindi dal

$$\max_{1 \leq i \leq n-1} |\frac{\lambda_{i+1}}{\lambda_i}|.$$

4.6 Shifting

Aumentiamo la velocità di convergenza con una tecnica di *shifting*: sia μ un numero che approssima λ, e consideriamo

$$\begin{cases} A_k - \mu I & = Q_k R_k \\ A_{k+1} & = R_k Q_k + \mu I \end{cases}$$

Si verifica facilmente che la successione di queste matrici ha gli autovalori invarianti.
Il modo migliore per accelerare maggiormente è scegliere μ che approssima λ_n.

1. Faccio r passi del metodo QR;
2. $\mu = a^{(r)}_{nn}$;
3. Opero con metodo QR con shift.

oppure si può cambiare μ ad ogni passo ponendolo $\mu_k = a^{(k)}_{nn}$. Si dimostra che in caso di matrici hermitiane questa seconda strategia permette di azzerare a_{n-1n} con ordine di convergenza 3.

Oss. 13: Oss.. In realtà opreo in maniera selettiva sempre sull'ultimo autovalore e poi riduco la matrice di una dimensione.

Se $|\lambda_{n-1}| = |\lambda_n|$ (per esempio se sono proprio uguali) allora:

$$A^{(k)}_{n-1} = \begin{pmatrix} a^{(k)}_{n-1n-1} & a^{(k)}_{n-1n} \\ a^{(k)}_{nn-1} & a^{(k)}_{nn} \end{pmatrix}$$

e gli autovalori di questa matrice approssimano λ_n dunque li uso come parametro μ.

Oss. 14: Oss.. Lavorare in aritmetica complessa comporta un aggravo del costo computazionale.

4.6.1 Metodo QR con doppio shift

Consideriamo α e β che approssimano gli autovalori di A_{n-1} (cioè $\lambda_{n-1} = \lambda_n$). Definiamo:

$$\begin{aligned} A_k - \alpha I &= Q_k R_k \\ A_{k+1} &= R_k Q_k + \alpha I \\ A_{k+1} - \beta I &= Q_{k+1} R_{k+1} \\ A_{k+2} &= R_{k+1} Q_{k+1} + \beta I \end{aligned}$$

e posti $S = R_{k+1}R_k$ e $Z = Q_k Q_{k+1}$ che sono rispettivamente triangolare superiore ed unitaria si verifica che

$$ZS = A_k^2 - (\alpha + \beta)A_k + \alpha\beta I =: M$$

è a elementi reali, quindi la fattorizzazione QR ZS è a elementi reali. Inoltre si ottiene che $A_{k+2} = Z^H A_k Z$. In pratica invece di operare il doppio shift posso costruire M, trovare Z fattorizzando e operare trasformazione di similitudine. Tuttavia in questo modo, calcolando A_k^2 potrei distruggere la struttura e aumentare il costo computazionale ($O(n^3)$ che si può limitare col *teorema del Q implicito*).

Teorema 4.9 (del Q implicito)**.** *Sia $A = QHQ^T$ e $A = VGV^T$ dove H e G sono in forma di Hessenberg superiore ed irriducibili. $Q = \begin{pmatrix} q_1 & \cdots & q_n \end{pmatrix}$ e $V = \begin{pmatrix} v_1 & \cdots & v_n \end{pmatrix}$ ortogonali con $q_1 = v_1$. Allora $\forall i = 2, \dots, n$ $q_i = \pm v_i$.*

$\diamondsuit$

4.6.2 Bulge - Chasing

Questo algoritmo è stato introdotto da Francis nel 1961.

- Calcoliamo la prima colonna di M:

$$Me_1 = \begin{pmatrix} a_{11}^2 + a_{12}a_{21} - (\alpha+\beta)a_{11} + \alpha\beta \\ a_{11}a_{21} + a_{21}a_{22} - (\alpha+\beta)a_{21} \\ a_{32}a_{21} \\ 0 \\ \vdots \end{pmatrix}$$

- Costruiamo una matrice P_0 di Householder tale che $P_0 M e_1 = \gamma e_1$ (in costo $O(1)$);
- Calcoliamo $P_1, \ldots, P_{n-2}$ di Householder per cui $Z' := P_0 \cdots P_{n-2}$ ha come prima colonna la prima colonna di P_0;
- $A_3 = Z'^H A_1 Z'$ sfruttando il teorema del Q *implicito* e bypassando la fattorizzazione di M.
- $\widetilde{P_0}$ di Householder di ordine 3, P_0 =diag$(\widetilde{P_0}, I_{n-3})$;
- moltiplicando $P_0 A_1 P_0$ si inseriscono 3 elementi in posizione $(3,1)$, $(4,1)$ $(4,2)$;
- sia P_1 =diag$(1, \widetilde{P_1}, I_{n-4})$;
- il prodotto $P_1 P_0 A_1 P_0 P_1$ shifta gli elementi non nulli verso il basso;
- iterando si ottiene

$$P_{n-3} \cdots P_0 A_1 P_0 \cdots P_{n-3} = \begin{pmatrix} * & \cdots & & & \cdots \\ * & \ddots & & & \\ & & \ddots & \ddots & \\ 0 & \cdots & + & * & * \end{pmatrix}$$

  dove $+$ è in posizione $(n, n-2)$;
- $P_{n-2} = \begin{pmatrix} c & -s \\ s & c \end{pmatrix}$ matrice di Givens che annulla anche l'ultimo elemento per ottenere una forma di Hessenberg superiore;
- $Z'^H = P_{n-2} \cdots P_0$ ha per prima colonna la stessa di Z per come ho costruito le P_i;

4.7 Metodo delle potenze

Oss. 15: Info a caso.

- Dal 1913 by Müntz;
- si applica a matrici qualunque;

Supponiamo A diagonalizzabile con autovalori $|\lambda_1| > |\lambda_2| \geq |\lambda_3| \geq \cdots \geq |\lambda_n|$ (quindi $\lambda_1 \in \mathbb{R}$ ed è detto autovalore dominante); siano $x_1, \ldots, x_n$ i rispettivi autovettori linearmente indipendenti. Sia $t_0 \in \mathbb{C}^n$. Allora $t_0 = \sum_{i=0}^{n} \alpha_i x_i$ con $\alpha_1 \neq 0$. Definiamo una successione

$$y_0 = t_0 \quad y_k = Ay_{k-1} \quad \text{per } k = 1, 2, \ldots$$

e cioè si ha

$$\begin{aligned} y_k &= A^k y_0 = A^k t_0 \\ &= \lambda_1^k \left[\alpha_1 x_1 + \sum_{i=1}^{n} \alpha_i (\frac{\lambda_i}{\lambda_1})^k x_i \right] \end{aligned}$$

e prendendo il rapporto tra due componenti consecutive (non nulle):

$$\lim_{k \to +\infty} \frac{(y_{k+1})_j}{(y_k)_j} = \lambda_1$$

e inoltre $\lim_{k \to +\infty} \frac{y_k}{\lambda_1^k} = \alpha_1 x_1$.

Oss. 16: Oss..

- Per matrici generiche il costo della generazione di y_k è $O(n^2)$.
- Se $|\lambda_1| \neq 1$ il processo potrebbe creare *overflow* o *underflow*.

Ora normalizzo i vettori della successione:

$$u_k = A_k t_{k-1} \quad t_k = \frac{1}{\beta_k} u_k \; k = 1, 2, \ldots$$

dove β_k è tale che $\|t_k\| = 1$.

Oss. 17: !?. Quale norma conviene prendere?

Ricorsivamente si ottiene

$$t_k = \frac{1}{\prod_{i=1}^{k} \beta_i} A^k t_0 = \frac{1}{\gamma_k} A^k t_0 u_{k+1} = A t_k = \frac{1}{\gamma_k} A^{k+1} t_0$$

e considero il rapporto delle j-esime componenti che per $k \to +\infty$ tende a λ_1.

Oss. 18: $\|\cdot\|_\infty$.*** conti *** si sceglie come criterio di arresto

$$|\beta_{k+1} - \beta_k| < \varepsilon$$

Oss. 19: $\|\cdot\|_2$.Questa è più dispendiosa da calcolare, ma se A è hermitiana conviene. *** conti *** si sceglie come criterio d'arresto

$$\|u_{k+1} - \sigma_k t_k\|_2 < \varepsilon \quad (\sigma_k = t_k^H u_{k+1})$$

4.8 Metodo delle potenze inverse

Osserviamo che se gli autovalori di A sono $|\lambda_1| \geq |\lambda_2| \geq \cdots > |\lambda_n|$ posso applicare il metodo delle potenze a A^{-1}. Ma dal punto di vista numerico è sconveniente calcolare l'inversa di una matrice.
É possibile risolvere il sistema lineare

$$Au_k = t_{k-1} \quad t_k = \frac{1}{\beta_k} u_k$$

e poiché la matrice non cambia, fattorizzandola la prima volta accelero la risoluzione ad ogni passo: a parte il costo iniziale, ad ogni passo pago $O(n^2)$.

4.9 Metodo delle potenze inverse con shift

Supponiamo μ stima di λ_j tale che $|\mu - \lambda_j| < |\mu - \lambda_i| \forall i \neq j$. La matrice $A - \mu I$ ha gli autovalori come nelle ipotesi per applicare il metodo delle potenze inverse:

$$(A - \mu I)u_k = t_{k-1} \quad t_k = \frac{1}{\beta_k} u_k \quad \text{con} \beta_k \to \frac{1}{\lambda_j - \mu} * * * \pm ? * **$$

Se x è autovettore associato a λ_j, $Ax = \lambda_j x \iff (A - \mu I)x = Ax - \mu x = (\lambda_j - \mu)x$ è autovettore associato a $\lambda_j - \mu$.

Oss. 20: **Non.**vogliamo che μ sia una stima troppo precisa: infatti se $\mu - \lambda_j \sim 0$ allora A è quasi singolare e il problema diventa mal condizionato.

4.10 Algoritmo QZ

Oss. 21: Info a caso.

- Dal 1973 by Moler & Steward;

Se Z è una matrice unitaria, poiché $Ax = \lambda Bx$ corrisponde a $QAx = \lambda QBx$ allora $QAZy = \lambda QBZy$ con $Zy = x$. Gli autovalori non sono cambiati, gli autovettori invece sì tramite trasformazione unitaria.

1. A si porta in forma di Hessenberg superiore, mentre B in forma triangolare superiore.
2. Se B è non singolare si può calcolare $C = AB^{-1}$.
3. Si applica un passo di QR con shift:

$$\begin{aligned}(C - \sigma I) = Q^T R \Rightarrow R = Q(C - \sigma I)\\ C' = RQ^T + \sigma I = QCQ^T \text{ (Hessenberg superiore)}\\ A' = QAZ \quad B' = QBZ\\ A'B'^{-1} = C'\end{aligned}$$

   (con Z unitaria costruita in modo che B' sia triangolare superiore). Segue dall'ultima uguaglianza che A' è in forma di Hessenberg superiore.

4. $$Q(AB^{-1} - \sigma I) = R \Rightarrow Q(A - \sigma B) = RB =: S \text{ (triangolare superiore)}$$

   e cioè Q può essere di Householder.

   Osservazione 22. In realtà non conviene, $A - \sigma B$ è quasi triangolare superiore, Q è una matrice con blocchetti 2×2 di Householder che annulla elementi specifici sotto la diagonale, similmente al QR con doppio shift:

$$Q = Q_{n-1} \cdots Q_2 Q_1 \quad Q_j = \begin{pmatrix} I & & \\ & \widetilde{Q_j} & \\ & & I \end{pmatrix}$$

   dove $\widetilde{Q_j}$ è il blocchetto 2×2.

5. Per costruire Z utilizzo un metodo simile al *Bull chasing*: Moltiplicando $Q_1 B$ costruisco Z_1 in modo che $Q_1 B Z_1$ sposti il problema alla riga dopo.

6. Ora $QBZ_1 \cdots Z_{n-1} = QBZ$ è triangolare superiore.

7. $A' = QAZ = SZ + \sigma QBZ$ essendo somma di una Hessenberg superiore e di una triangolare superiore è una Hessenberg superiore.

8. Iterando il ragionamento ottengo due successioni A_ν e B_ν tali per cui $C_\nu = A_\nu B_\nu^{-1} \to \mathbf{T}$ triangolare superiore. (è come fare il QR su C_ν). Da questo segue che anche la successione $A_\nu \to A_T$ triangolare superiore (essendo le B_ν triangolari superiori).

5 Problema lineare dei minimi quadrati

Siano $A \in \mathbb{C}^{m\times n}$, $b \in \mathbb{C}^m$. Supponiamo di voler trovare $x \in \mathbb{C}^n$ tale che $Ax = b$. Se $m \geq n$ il problema è sovrdeterminato ed ammette soluzione se e solo se $b \in S(A) = \{y \in \mathbb{C}^n : y = Ax,\, x \in \mathbb{C}^n\}$. Altrimenti il problema diventa trovare

$$x \in \mathbb{C}^n \quad \|Ax - b\|_2 = \min_{y\in\mathbb{C}} \|Ay - b\|_2 =: \gamma$$

ovvero minimizzare la norma del residuo. Tale problema è detto problema lineare dei minimi quadrati.

Consideriamo $S(A)^\perp = \{z \in \mathbb{C}^n : z^H y = 0\, \forall y \in S(A)\}$ e decomponiamo $b = b_1 + b_2$ con $b_1 \in S(A)$ e $b_2 \in S(A)^\perp$. Allora

$$r = b - Ax = b_1 + b_2 - Ax = y + b_2 \quad \text{con } y \in S(A)$$

pertanto per minimizzare

$$\|r\|_2^2 = r^H r = \|y\|_2^2 + \|b_2\|_2^2$$

bisogna minimizzare $\|y\|_2$ e quindi risolvere il sistema lineare $Ax = b_1$. Se $y = 0$ si ha la catena di implicazioni:

$$r = b_2 \iff r \in S(A)^\perp \iff A^H r = 0 \iff A^H Ax = A^H b$$

e cioè x è soluzione del problema dei minimi quadrati se e solo se x è soluzione del sistema delle **equazioni normali**.

Osservazione 1. Notiamo che $A^H A$ è simmetrica e definita positiva, quindi si può usare la fattorizzazione di Cholesky e trovare L triangolare inferiore per cui $A^H A = LL^H$ e risolvere

$$\begin{cases} Ly = & A^H b \\ L^H x = & y \end{cases}$$

ma si hanno due problemi:

1) Il problema $A^H Ax = A^H b$ può essere fortemente mal condizionato.

2) Con numeri troppo piccoli nel calcolatore si hanno problemi, ad esempio:

$$A = \begin{pmatrix} 1 & 1 \\ \alpha & 0 \\ 0 & \alpha \end{pmatrix}$$

dove la precisione di macchina è $\mathbf{u} = 10^{-16}$ ed $\alpha = 10^{-10}$ porta a

$$A^H A = \begin{pmatrix} 1+\alpha^2 & 1 \\ 1 & 1+\alpha^2 \end{pmatrix} = \begin{pmatrix} 1 & 1 \\ 1 & 1 \end{pmatrix}$$

essendo $\alpha^2 = 10^{-20} < \mathbf{u}$ e quindi $A^H A$ ha rango 1, non è definita positiva!

5.1 Metodo QR

Oss. 2: Attenzione. Il nome può creare confusione: questo è il metodo per la risoluzione di sistemi lineari, c'è l'omonimo per la stima degli autovalori.

Sia $A \in \mathbb{C}^{m\times n}$ di rango massimo n (supponendo $m \geq n$). Attraverso matrici di Householder posso trasformare

$$A = QR = \left(\begin{array}{cc} a & \\ & * \\ 0 & b \\ \hline & \\ 0 & \end{array}\right) \qquad R = \begin{pmatrix} R_1 \\ 0 \end{pmatrix} \in \mathbb{C}^{n+(n-m)\times n}$$

[ultra thick]ab dove $Q \in \mathbb{C}^{m\times n}$ è unitaria e $\det R_1 \neq 0$.

$$\|Ax - b\|_2 = \|QRx - b\|_2 = \|Q(Rx - Q^T b)\|_2 = \|Rx - Q^T b\|_2$$

e posto $c := Q^T b = \begin{pmatrix} c_1 \\ c_2 \end{pmatrix} \in \mathbb{C}^{n+(m-n)}$ il problema da risolvere diventa:

$$\min_{y\in\mathbb{C}^n} \|Ay - b\|_2^2 = \min_{y\in\mathbb{C}^n} (\|R_1 y - c_1\|_2^2 + \|c_2\|_2^2) = \|c_2\|_2^2 + \min_{y\in\mathbb{C}^n} \|R_1 y - c_1\|_2^2$$

e si tratta di risolvere $R_1 y = c_1$.

Oss. 3: Oss..A meno di matrici di fase

$$LL^H = A^H A = R^H Q^H Q R = R^H R = R_1^H R_1$$

e cioè $R_1 = L^H$.

5.2 SVD

Teorema 5.1. *Sia $A \in \mathbb{C}^{m\times n}$. Esistono $U \in \mathbb{C}^{m\times m}$ e $V \in \mathbb{C}^{n\times n}$ unitarie per cui $A = U\Sigma V^H$ dove*

$$\Sigma = \begin{pmatrix} \sigma_1 & & \\ & \ddots & \\ & & \sigma_p \\ \hline & 0 & \end{pmatrix}$$

$p = \min\{m, n\}$ *e*

$$\Sigma_{ij} = \begin{cases} 0 \; i \neq j \\ \sigma_i \; i = j \end{cases} \qquad \sigma_1 \geq \sigma_2 \geq \ldots \sigma_p \geq 0$$

$\diamondsuit$

Definizione 5.1. Tale decomposizione di A è detta **decomposizione ai valori singolari** di A (***SVD***); i σ_i si dicono valori singolari e le colonne

$$U = \begin{pmatrix} u_1 & \cdots & u_m \end{pmatrix} \quad V = \begin{pmatrix} v_1 & \cdots & v_n \end{pmatrix}$$

si dicono vettori singolari rispettivamente sinistri e destri di A.

Teorema 5.2. *Sia $A \in \mathbb{C}^{m\times n}$ con SVD $A = U\Sigma V^H$ dove $\sigma_1 \geq \cdots \geq \sigma_k > \sigma_{k+1} = \cdots = \sigma_p = 0$, con $p = \min\{m, n\}$. Allora*

a) $A = U_k\Sigma_k V_k^H = \sum_{i=1}^{k} \sigma_i u_i v_i^H$ *essendo*

$$U_k = (u_1 \cdots u_k) \in \mathbb{C}^{m\times k} \quad V = (v_1 \cdots v_k) \in \mathbb{C}^{n\times k} \quad \Sigma_k = diag(\sigma_i) \in \mathbb{R}^{k\times k}$$

b) $N(A) = Ker(A) = Span(v_{k+1}, \ldots, v_n)$

c) $S(S) = Span(u_1, \ldots, u_k)$ *e quindi* $rkA = k$.

d) σ_i^2 *sono autovalori di* $A^H A$ *da cui* $\|A\|_2 = \sigma_1^2$

$\diamondsuit$

Oss. 4: Oss.. La SVD ci permette di studiare teoricamente $N(A)$, $S(A)$ (e $N(A^T)$, $S(A^T)$). I punti *a*) e *c*) implicano che una matrice di rango k si può pensare come combinazione lineare di matrici di rango 1.

5.3 Come ottenere una SVD

i) Calcolare $A^H A$;

ii) $A^H A = QDQ^H$ in forma di Schur con D diagonale e Q unitaria (ordinando per convenzione gli autovalori di D in modo non crescente);

iii) $C = AQ \in \mathbb{C}^{m\times n}$, ne permuto le righe tramite Π e fattorizzo QR:

$$C\Pi = UR = U\begin{pmatrix} R_1 \\ 0 \end{pmatrix}$$

dove U è unitaria, R triangolare superiore con elementi non crescenti;

iv) $A = CQ^H = C\Pi\Pi^H Q^H = UR\Pi^H Q^H$ (le matrici di permutazione sono unitarie);

v) Si ha che:

$$\begin{aligned} A^H A =& Q\Pi R^H U^H U R \Pi^H Q^H \\ =& Q\Pi R_1^H R_1 \Pi^H Q^H \\ \Rightarrow QDQ^H =& Q\Pi R_1^H R_1 \Pi^H Q^H \\ \Rightarrow \Pi^H D\Pi =& R_1^H R_1 \\ \Rightarrow& R_1^H R_1 \text{ è diagonale} \end{aligned}$$

e quindi R_1 è diagonale;

vi) $A = UR\Pi^T Q^H = U\Sigma V^H$ è la SVD di A.

5.4 Calcolare la forma di Schur di $A^H A$

Bisogna trovare P e H unitarie tali che, se B è bidiagonale superiore di ordine n

$$PAH = \begin{pmatrix} B \\ 0 \end{pmatrix}$$

E quindi si ha:

$$A^H A = H\begin{pmatrix} B^T & 0 \end{pmatrix} PP^H \begin{pmatrix} B \\ 0 \end{pmatrix} H^H = H(B^T B)H^H$$

Notando che $B^T B$ è tridiagonale e simmetrica, $A^H A$ è simile ad una matrice tridiagonale. Applicando il metodo QR per il calcolo degli autovalori $B^T B = WDW^T$ dove $D = \text{diag}(\sigma_1^2, \ldots, \sigma_n^2)$.

$$A^H A = HWDW^T H^H \Rightarrow Q := HW \quad A^H A = QDQ^H$$

è la forma cercata. Quindi per ottenere Q servono W (metodo QR) e H (Golub Reinsch, 1970). Definiamo ora $A^{(1)} = A$ e costruiamo $P^{(1)}$ di Householder in modo che $A^{(2)} = P^{(1)}A^{(1)} = \begin{pmatrix} \alpha & c^H \\ 0 & B^{(2)} \end{pmatrix}$ dove $c \in \mathbb{C}^{n-1}$. Inoltre costruiamo $K^{(1)} \in \mathbb{C}^{(n-1)\times(n-1)}$ di Householder che annulli tutti gli elementi della prima riga dal terzo in poi.

$$H^{(1)} := \begin{pmatrix} 1 & \\ & K^{(1)} \end{pmatrix} \Rightarrow A^{(3)} = A^{(2)}H^{(1)} = \left(\begin{array}{c|cccc} * & * & 0 & \dots & 0 \\ \hline 0 & & & & \\ \vdots & & & & \\ 0 & & & & \end{array}\right)$$

Iterando per ogni $A \in \mathbb{C}^{m\times n}$ si ottiene

$$P^{(n)} \cdots P^{(1)} A H^{(1)} \cdots H^{(n-2)} = \begin{pmatrix} \dots \end{pmatrix}$$

Il costo totale è $2mn^2 - \frac{2}{3}n^3$ moltiplicazioni. Applicando il metodo QR a B^TB so ha costo $O(n)$.

5.5 Applicazione della SVD al problema lineare dei minimi quadrati

Teorema 5.3. *Sia $A \in \mathbb{C}^{m\times n}$ con $m \geq n \geq k$ con $k =$rk$A$ e $A = U\Sigma V^H$. Allora $x^* = \sum_{i=1}^{k} \frac{u_i^H b}{\sigma_i} v_i$ è la soluzione al problema.*

$\diamondsuit$

Definizione 5.2. Data $A \in \mathbb{C}^{n\times n}$ di rango k con SVD $A = U\Sigma V^H$. Definiamo $A^+ \in \mathbb{C}^{m\times n}$ come $A^+ = V\Sigma^+U^H$ dove

$$(\Sigma^+)_{ij} = \begin{cases} 1/\sigma_i \text{ per } i = j \in \{1, \dots, k\} \\ 0 \end{cases} .$$

A^+ è detta **pseudoinversa** (*di Moore-Penrose*) di A.

Oss. 5: Coerenza. Se $A \in \mathbb{C}^{n\times n}$ con $\det A \neq 0$ allora $A^+ = A^{-1}$.

Dunque possiamo esprimere la soluzione del problema dei minimi quadrati in termini di pseudoinversa:

$$x^* = A^+ b$$

Oss. 6: Condizionamento del problema delle equazioni normali.A^HA ha autovalori σ_1^2 e la sua pseudoinversa $(A^HA)^+$ ha valori singolari $1/\sigma_i^2$

per cui $\mu_2(A^HA) = \frac{\sigma_1^2}{\sigma_k^2} = \mu_2(A)^2$. Questo dice che se $\mu_2(A)$ è relativamente grande, il numero di condizionamento nel problema delle equazioni normali aumenta quadraticamente.

Sia $A \in \mathbb{C}^{n\times n}$ normale con $A = UDU^H = U|D|\operatorname{sgn}(D)U^H$ essendo

$$D = \begin{pmatrix} \lambda_1 & & \\ & \ddots & \\ & & \lambda_n \end{pmatrix}$$

Vale anche $A = \Sigma U V^H$. E $\sigma_i = |\lambda_i|$ per il legame tra autovalori e valori singolari. $A^HA = V\Sigma^T\Sigma V^H = U\Sigma\Sigma^T U^H = AA^H$. Infatti ad esempio se $m \geq n$

$$\Sigma\Sigma^T = \begin{pmatrix} \sigma_1^2 & & & \\ & \ddots & & \\ & & \sigma_n^2 & \\ & & & \mathbf{0} \end{pmatrix} \quad \text{e} \quad \Sigma^T\Sigma = \begin{pmatrix} \sigma_1 & & \\ & \ddots & \\ & & \sigma_n \end{pmatrix}$$

(in entrambi i casi gli autovalori sono σ_i^2).Gli autovettori sono in un caso le colonne di V e nell'altro quelle di U.

Teorema 5.4. *Sia $A \in \mathbb{C}^{n\times n}$. Allora per ogni autovalore λ di A vale $\sigma_n \leq |\lambda| \leq \sigma_1$.*

$\diamondsuit$

Oss. 7: Oss.. $\mu_2(A) = \frac{\sigma_1}{\sigma_n} \geq \frac{|\lambda_1|}{|\lambda_n|}$ essendo λ_1 l'autovalore di modulo massimo e λ_n di modulo minimo.

Nel caso di matrici normali il numero di condizionamento coincide con quello dato da $\|A\|\|A^{-1}\|$, altrimenti il problema puó essere mal condizionato nonostante il rapporto tra gli autovalori di modulo massimo e minimo sia piccolo.
Sia $A \in \mathbb{C}^{m\times n}$ di rango $k \leq n \leq m$ e sia $r < k$ intero positivo. Cerchiamo $B \in \mathbb{C}^{m\times n}$ *"vicina"* ad A.

Teorema 5.5. *Sia $A \in \mathbb{C}^{m\times n}$ con SVD $A = U\Sigma V^H$ e $\sigma_1 \geq \cdots \geq \sigma_r > \sigma_{r+1} = \cdots = \sigma_n = 0$ con $r \leq k$. Definiamo $A_r := \sum_{i=1}^{n} \sigma_i u_i v_i^H$ e $S := \{B \in \mathbb{C}^{m\times n} \mid rkB = r\}$. Allora A_r è la matrice di rango r piú vicina ad A:*

$$\min_{B\in S} \|A - B\|_2 = \|A - A_r\|_2 = \sigma_{r+1}$$

$\diamondsuit$

Oss. 8: Applicazioni del teorema. Sia $Ax = b$, $\mu(A) = \frac{\sigma_1}{\sigma_n} \simeq 10^{12}$ e $u = 10^{-16}$. Ci aspettiamo di avere una soluzione approssimata con al piú 4 cifre significative (esatte). La grandezza del numero di condizionamento potrebbe essere dovuta al fatto che σ_n sia di ordine di grandezza molto minore rispetto agli altri.
Invece di risolvere $Ax = b$ risolvo $\widetilde{A}x = b$ con valori signolari $\sigma_1, \ldots, \sigma_{n-1}, 0$ (ovvero scelgo un valore soglia sotto il quale i valori singolari vengono considerati nulli. Ad esempio Matlab nel calcolare il rango di una matrice e costruire la SVD utilizza come soglia $\varepsilon = \max\{m, n\}\sigma_1 u$

5.6 Metodo di Lanczos

***?? Data f e x_i per $i = 1, \dots, m$ a due a due distinti, si cerca $p(x) = \alpha_0 + \alpha_1 x + \cdots + \alpha_{n-1} x^{n-1}$ tale che $\sum_{i=1}^{m} (p(x_i) - f(x_i))^2$ è minima. Il polinomio p è il polinomio di approssimazione ai minimi quadrati di f.
Introduciamo la matrice

$$A = \begin{pmatrix} x_1^0 & \dots & x_1^{n-1} \\ \vdots & \ddots & \vdots \\ x_m^0 & \dots & x_m^{n-1} \end{pmatrix} \in \mathbb{R}^{m \times n}$$

e i vettori

$$y = \begin{pmatrix} \alpha_0 \\ \vdots \\ \alpha_{n-1} \end{pmatrix} \qquad b = \begin{pmatrix} f(x_1) \\ \vdots \\ f(x_n) \end{pmatrix}$$

(Nel caso $m = n$ si tratta di un problema di interpolazione con A matrice di Vandermonde). In genere A è sparsa e di grandi dimensioni: bisogna evitare il *fill in*. Notiamo che se $A \in \mathbb{R}^{m \times n}$, si puó costruire $B = \begin{pmatrix} 0 & A \\ A^T & 0 \end{pmatrix} \in \mathbb{R}^{(m+n) \times (m+n)}$ ed il problema di trovare valori e vettori singolari per A si riconduce a trovare autovalori ed autovettori per B. Possiamo scrivere $A = U \Sigma V^T$ con

$$U = \begin{bmatrix} U_1 & U_2 \end{bmatrix} \in \mathbb{R}^{m \times (n+(m-n))} \quad \Sigma = \begin{bmatrix} \Sigma_1 \\ 0 \end{bmatrix} \in \mathbb{R}^{(n+(m-n) \times m} \quad V \in \mathbb{R}^{n \times n}$$

e posto

$$Z = \frac{1}{\sqrt{2}} \begin{bmatrix} U_1 & U_1 & \sqrt{2} U_2 \\ V & -V & 0 \end{bmatrix} \in \mathbb{R}^{(m+n) \times (m+n)}$$

che è ortogonale, si ottiene

$$B = Z \begin{pmatrix} \Sigma_1 & & \\ & -\Sigma_1 & \\ & & 0 \end{pmatrix} Z^T$$

Oss. 9: Oss.. Nel caso $m = n$ si ha $Z = \frac{1}{\sqrt{2}} \begin{pmatrix} U & U \\ V & -V \end{pmatrix}$ e $z = \frac{1}{\sqrt{2}} \begin{pmatrix} u \\ v \end{pmatrix}$ con $\|u\|_2 = \|v\|_2 + 1$.

***??

Data $A \in \mathbb{C}^{m \times n}$ con $A = A^H$, troviamo $Q = \begin{pmatrix} q_1 \dots q_n \end{pmatrix}$ unitaria, assegnato q_1 tale che $Q^H A Q = T =$tridiag$(\beta_i, \alpha_j, \beta_i)$. Partiamo da $AQ = QT$ e

guardiamo l'equazione per colonne:

$$\begin{aligned} Aq_1 &= \alpha_1 q_1 + \beta_1 q_2 \\ &\dots \\ Aq_i &= \beta_{i-1} q_{i-1} + \alpha_i q_i + \beta_i q_{i+1} \\ &\dots \\ Aq_n &= \beta_{n-1} q_{n-1} + \alpha_n q_n \end{aligned}$$

e ricaviamo, imponendo $q_1^H q_2 = 0$ e sapendo $\|q_1\|_2 = 1$,

$$q_1^H A q_1 = \alpha_1 q_1^H q_1 + \beta_1 q_1^H q_2 = \alpha_1 q_2 = \frac{(A - \alpha_1 I) q_1}{\beta_1}$$

e ancora, imponendo $\|q_2\|_2 = 1$ si ottiene $\beta_1 = \|A - \alpha I\|_2$. Iterando il processo si ottiene

$$\begin{aligned} \alpha_i &= q_i^H A q_i \\ q_{i+1} &= \frac{(A - \alpha_i I) q_i - \beta_{i-1} q_{i-1}}{\beta_i} \\ \beta_i &= \|(A - \alpha_i I) q_i - \beta_{i-1} q_{i-1}\|_2 \\ \alpha_n &= q_n^H A q_n \end{aligned}$$

Questo funziona se $\forall i \, \beta_i > 0$. Nel caso in cui qualcuno sia nullo bisogna trovare un metodo alternativo.
I vettori q_i si dicono *vettori di Lanczos.*
Applicando il metodo alla matrice $B = \begin{pmatrix} 0 & A \\ A^T & 0 \end{pmatrix}$ si ottiene una matrice tridiagonale di cui sappiamo calcolare autovalori ed autovettori con le tecniche dei capitoli precedenti. É possibile scegliere q_1 in modo da ottimizzare l'occupazione di memoria. Si ha infatti:

Teorema 5.6 (Golub, Kahan, 1965)**.** *Sia $A \in \mathbb{R}^{m \times n}$ con $m \geq n$ e sia $B = \begin{pmatrix} 0 & A \\ A^T & 0 \end{pmatrix} \in \mathbb{R}^{(m+n)\times(m+n)}$. Applicando a B il metodo di Lanczos con $q_1 = \begin{pmatrix} u \\ 0 \end{pmatrix}$ oppure con $q_1 = \begin{pmatrix} 0 \\ v \end{pmatrix}$ dove $u \in \mathbb{R}^m$ e $v \in \mathbb{R}^n$ di norma $\|u\|_2 = \|v\|_2 = 1$ si risparmia tempo e memoria.*

♢

Osservazione 10. Ci potrebbero essere problemi per il calcolo degli autovalori di modulo più piccolo.

6 Risoluzione di sistemi lineari di grosse dimensioni

Oss. 1: Oss.. Il metodo di Gauss su una matrice sparsa causa *fill in* (e in generale questo vale per i metodi diretti).

Conviene utilizzare i **metodi iterativi** (Jacobi, Gauss-Seidel, ...):

a) $A \in \mathbb{C}^{n\times n}$ con $\det A \neq 0$;

b) Splitting: $A = M - N$ con $\det M \neq 0$;

c) $Ax = b \iff Mx = Nx + b \iff x = M^{-1}Nx + M^{-1}b$;

d) $P := M^{-1}N$ e $q := M^{-1}b$ da cui $x = Px + q$;

e) Dato un vettore iniziale $X^{(0)}$ si costruisce la successione $x^{(k)} = Px^{(k-1)} + q$

D'ora in avanti si utilizzerà il seguente splitting:

$$A = D-B-C = \begin{pmatrix} a & & \mathbf{0} \\ & & \\ \mathbf{0} & & b \end{pmatrix} - \begin{pmatrix} 0 & c & & \\ & \ddots & & * \\ & & \ddots & d \\ \mathbf{0} & & & 0 \end{pmatrix} - \begin{pmatrix} 0 & & & \\ e & \ddots & & \mathbf{0} \\ & & \ddots & \\ & * & & f \quad 0 \end{pmatrix}$$

[ultra thick]ab [ultra thick]cd [ultra thick]ef ovvero

$$d_{ij} = \begin{cases} a_{ii} & i = j \\ 0 & i \neq j \end{cases} \quad b_{ij} = \begin{cases} -a_{ij} & i > j \\ 0 & i \leq j \end{cases} \quad c_{ij} = \begin{cases} -a_{ij} & i < j \\ 0 & i \geq j \end{cases}$$

In base alle combinazioni per scegliere M ed N si ottengono:

- Metodo di JACOBI
  - $M = D$
  - $N = B + C$
  - e matrice di iterazione $J = D^{-1}(B + C)$
- Metodo di GAUSS-SEIDEL
  - $M = D - B$
  - $N = C$
  - e matrice di iterazione $G = (D - B)^{-1}C$
- Metodo di RILASSAMENTO
  - $M = D - \omega B$

- $N = (1-\omega)D + \omega C$
- e matrice di iterazione $H(\omega) = (D - \omega B)^{-1}[(1-\omega)D + \omega C]$

Osservazione 2. Il metodo di rilassamento si ottiene considerando il sistema $\omega Ax = \omega b$ dove $\omega \in \mathbb{C}$ e $\omega \neq 0$. Svolgendo un po' di conti si arriva a scrivere

$$x_i^{(k)} = (1-\omega)x_i^{(k-1)} + \frac{\omega}{a_{ii}}\left[b_i - \sum_{j=1}^{i-1} a_{ij}x_j^{(k)} - \sum_{j=i+1}^{n} a_{ij}x_j^{(k-1)}\right]$$

Oss. 3: Oss.. Il metodo di rilassamento con $\omega = 1$ è quello di Gauss-Seidel e $H(1) = G$.

Definizione 6.1. Se $\omega < 1$ il metodo si dice di *sottorilassamento*;
se $\omega > 1$ il metodo si dice si *sovrarilassamento* o *SOR* (Successive Over-Relaxation)

Teorema 6.1 (Kahan)**.** *$\rho(H(\omega)) \geq |\omega - 1|$ e una condizione necessaria alla convergenza è $|\omega - 1| < 1$. In particolare se $\omega \in \mathbb{R}$ la condizione è che sia $\omega \in (0, 2)$.*

♢

Teorema 6.2 (Ostrowski-Reich)**.** *Sia $A \in \mathbb{C}^{n\times n}$ definita positiva e simmetrica, e sia $\omega \in \mathbb{R}$ con $0 < \omega < 2$. Allora il metodo di rilassamento converge.*

♢

Teorema 6.3. *Sia $A \in \mathbb{C}^{n\times n}$ tridiagonale con $0 < \omega < 2$. Allora*

a) Se μ è autovalore di J, ogni λ tale che $(\lambda+\omega-1)^2 = \lambda\omega^2\mu$ è autovalore di $H(\omega)$;

b)

c) Se J ha autovalori reali $\rho(J) < 1$ allora $\exists!\omega_0$ tale che $\rho(H(\omega_0)) = \min\limits_{0<\omega<2} \rho(H(\omega))$ ed $\omega_0 = \frac{2}{1+\sqrt{1-\rho(J)^2}}$

Esempio Consideriamo $A = \begin{pmatrix} 2 & -1 & & \\ -1 & \ddots & \ddots & \\ & \ddots & \ddots & -1 \\ & & -1 & 2 \end{pmatrix} \in \mathbb{R}^{6\times 6}$. Allora si ha $\rho(J) = 0.9009688$ e $\rho(G) = \rho(J)^2 = 0.8117447$ e gli autovalori di A sono reali. Risulta $\omega_0 = 1.394812$ che implica $\rho(H(\omega_0)) = 0.3949117 \approx \rho(G)^{4.5} (\approx \rho(J)^9)$.

7 Metodi di Krylov

Sono usati sia per la risoluzione di sistemi lineari sia per la soluzione di problemi agli autovalori. Dati A, b e un algoritmo capace di calcolare il prodotto matrice vettore, cerchiamo di risolvere il sistema lineare $Ax = b$. Definiamo

$$y_1 = b \quad y_2 = Ab \quad \dots \quad y_n = Ay_{n-1}$$

e poniamo $K = \left[\begin{array}{cccc} y_1 & y_2 & \cdots & y_n \end{array} \right]$. Osserviamo che

$$\begin{aligned} AK &= \left[\begin{array}{cccccc} Ay_1 & Ay_2 & \cdots & Ay_{n-1} & Ay_n \end{array} \right] \\ &= \left[\begin{array}{cccccc} y_2 & y_3 & \cdots & y_n & A^n y_1 \end{array} \right] \end{aligned}$$

e che le prime $n-1$ colonne di AK sono le ultime $n-1$ colonne di K. Supponiamo $\det K \neq 0$ e $c = -K^{-1}A^n y_1$, allora $AK = K \left[\begin{array}{cccc} e_2 & e_3 & \cdots & e_n & -c \end{array} \right]$ e

$$K^{-1}AK = \left(\begin{array}{cccc|c} 0 & & \cdots & 0 & -c_1 \\ 1 & 0 & & \vdots & \vdots \\ & \ddots & \ddots & & \\ & & \ddots & & \vdots \\ 0 & & & 1 & -c_n \end{array} \right)$$

è una matrice compagna in forma di Hessenberg superiore il cui polinomio caratteristico è $p(x) = x^n + c_n x^{n-1} + \cdots + c_2 x + c_1$, che è lo stesso della matrice A. Il problema grosso rimane il calcolo di c. Dato quello, dopo la trasformazione per similitudine la risoluzione del sistema lineare e il calcolo degli autovalori sono (o dovrebbero essere) facili. Per la discussione fatta sul metodo delle potenze, le colone di K tendono ad essere parallele e cioè K tende ad essere mal condizionata. (y_n tende all'autovettore relativo all'autovalore di modulo massimo). Ora fattorizzando $K = QR$ si ottiene $K^{-1}AK = R^{-1}Q^T AR =: C$ da cui $H := Q^T AQ = RCR^{-1}$ in forma di Hessenberg superiore.

Oss. 1: Oss.. Se A è simmetrica lo è anche H che quindi è tridiagonale (simmetrica).

Se $Q = \left[\begin{array}{ccc} q_1 & \cdots & q_n \end{array} \right]$ poiché $AQ = QH$ si scrive $Aq_j = \sum_{i=1}^{j+1} q_i h_{ij}$.

- Consideriamo $1 \leq m \leq j$: $q_m^T(Aq_j) = \sum_{i=1}^{j+1} q_m^T q_i h_{ij} = h_{mj}$.
- inoltre $h_{j+1j} q_{j+1} = Aq_j - \sum_{i=1}^{j} q_i h_{ij}$

e si ottiene

Oss. 2: Algortimo di Arnoldi.

$$
\begin{aligned}
&q_1 = b/\|b\|_2 \\
&\textbf{for } j = 1 : k \\
&\qquad z = Aq_j \\
&\qquad \textbf{for } i = 1 : j \\
&\qquad\qquad h_{ij} = q_i^T z \\
&\qquad\qquad z = z - h_{ij} q_i \\
&\qquad \textbf{end} \\
&\qquad h_{j+1j} = \|z\|_2 \\
&\qquad\quad \textbf{if } h_{j+1j} = 0 \ \boldsymbol{STOP} \\
&\qquad q_{j+1} = z/h_{j+1j} \\
&\qquad \textbf{end}
\end{aligned}
$$

I vettori q_j così calcolati sono detti *vettori di Arnoldi*. Se ci arrestiamo con $k < n$, dette

$$Q_k = \begin{bmatrix} q_1 & \cdots & q_k \end{bmatrix} \quad Q_u = \begin{bmatrix} q_{k+1} & \cdots & q_n \end{bmatrix} \quad Q = \begin{bmatrix} Q_k & Q_u \end{bmatrix}$$

l'algoritmo di Arnoldi fornisce Q_k e la prima colonna di Q_u.

$$H = Q^T A Q = \begin{pmatrix} Q_k^T A Q_k & Q_k^T A Q_u \\ Q_u^T A Q_k & Q_u^T A Q_u \end{pmatrix} = \begin{pmatrix} H_k & H_{uk} \\ H_{ku} & H_u \end{pmatrix}$$

dove il blocco H_k è Hessenberg superiore e l'unico elemento (eventualmente) diverso da 0 del blocco H_{ku} è quello in angolo alto a destra. Applicando Arnoldi con k passi conosciamo entrambi i blocchi H_k e H_{ku} perché l'elemento suddetto è dato dal prodotto della prima riga di Q_u^T (cioè q_{k+1}) e dell'ultima colonna di AQ_k (che pure conosciamo).

Oss. 3: Oss.. Nel caso A sia simmetrica l'algoritmo è quello di Lanczos e conosciamo anche il blocco H_{uk}; (i vettori q_j sono i vettori di Lanczos).

Oss. 4: Oss.. Abbiamo costruito una base ortonormale per lo spazio di **Krylov**

$$\mathcal{K}_k(A, b) = \mathrm{Span}(\{b, Ab, \ldots, A^{k-1}b\}).$$

L'idea è di riuscire a risolvere il problema usando l'algoritmo con un valore $k << n$. Cerchiamo un'approssimazione della soluzione utilizzando i vettori di Arnoldi.

$$x_k = \sum_{j=1}^{k} z_j q_j = Q_k z$$

dove la scelta di z è fatta in modo che

(a) $\|x_k - x\|_2$ sia minima; (*ma non conoscendo x quest'idea è impraticabile*)

(b) $\|r_k\|_2$ sia minima:

- se A è simmetrica abbiamo il metodo **MINRES** (*minimum residual*)
- se A non è simmetrica abbiamo il **GMRES** (*generalised minimum residual*)

(c) $r_k \perp \mathcal{K}_k(A, r_0)$ (e abbiamo il *metodo di Arnoldi* o **FOM**);

(d) (se A è definita positiva) $\|r_k\|_{A^{-1}}$ sia minima (e troviamo il **CG**).

Teorema 7.1. *Sia A simmetrica, $T_k = Q_k^T A Q_k$ tridiagonale e $r_k = b - Ax_k$.*

- *Se $\det T_k \neq 0$ e $x_k = Q_k T_k^{-1} e_1 \|b\|_2$ allora $Q_k^T r_k = 0$;*
- *se inoltre A è definita positiva, allora $\det T_k \neq 0$ e la scelta di x_k minimizza $\|r_k\|_{A^{-1}}$. Inoltre $r_k = \pm \|r_k\|_2 q_{k+1}$.*

$\diamondsuit$

7.1 CG come metodo di Krylov

Se A è definita positiva anche $T_k = Q_k^T A Q_k$ lo è, quindi esiste la fattorizzazione di Cholesky

$$T_k = \hat{L}_k \hat{L}_k^T = L_k D_k L_k^T$$

in cui L_k ha diagonale unitaria. Per il teorema precedente, $x_k = Q_k T_k^{-1} e_1 \|b\|_2$ si può riscrivere come:

$$\begin{aligned} x_k &= Q_k (L_k^{-T} D_k^{-1} L_k^{-1}) e_1 \|b\|_2 \\ &= (Q_k)(L_k^{-T})(D_k^{-1} L_k^{-1} e_1 \|b\|_2) \\ &=: \widetilde{P}_k y_k \end{aligned}$$

Proposizione 7.2. *Le colonne $\widetilde{p}_i$ di $\widetilde{P}_k = \begin{bmatrix} \widetilde{p}_1 & \cdots & \widetilde{p}_k \end{bmatrix}$ sono A-coniugate.*

Dimostrazione.

$$\widetilde{P}_k^T A \widetilde{P}_k = L_k^{-1} Q_k^T A Q_k L_k^{-T} = L_k^{-1} T_k L_k^{-T} = D_k$$

□ Verifichiamo che $y_k = \begin{pmatrix} y_{k-1} \\ \eta_k \end{pmatrix}$ e che $\widetilde{P}_k = \begin{pmatrix} \widetilde{P}_{k-1} & \widetilde{p}_k \end{pmatrix}$ in modo che risulta

$$\begin{aligned} x_k &= \widetilde{P}_k y_k \\ &= \begin{pmatrix} \widetilde{P}_{k-1} & \widetilde{p}_k \end{pmatrix} \begin{pmatrix} y_{k-1} \\ \eta_k \end{pmatrix} \\ &= \widetilde{P}_{k-1} y_{k-1} + \eta_k \widetilde{p}_k \\ &= x_{k-1} + \eta_k \widetilde{p}_k \end{aligned}$$

Infatti se

$$\begin{aligned} T_k &= \begin{pmatrix} \alpha_1 & \beta_1 & & \\ \beta_1 & \ddots & \ddots & \\ & \ddots & \ddots & \beta_{k-1} \\ & & \beta_{k-1} & \alpha_k \end{pmatrix} \\ &= L_k D_k L_K^T \\ &= \begin{pmatrix} 1 & & & \\ l_1 & \ddots & & \\ & \ddots & \ddots & \\ & & l_{k-1} & 1 \end{pmatrix} \begin{pmatrix} d_1 & & & \\ & \ddots & & \\ & & \ddots & \\ & & & d_k \end{pmatrix} \begin{pmatrix} 1 & & & \\ l_1 & \ddots & & \\ & \ddots & \ddots & \\ & & l_{k-1} & 1 \end{pmatrix}^T \\ &= \begin{pmatrix} L_{k-1} & \\ l_k e_{k-1}^T & 1 \end{pmatrix} \begin{pmatrix} D_{k-1} & \\ & d_k \end{pmatrix} \begin{pmatrix} L_{k-1} & \\ l_k e_{k-1}^T & 1 \end{pmatrix}^T \end{aligned}$$

Poiché

$$D_k^{-1} = \begin{pmatrix} D_{k-1}^{-1} & \\ & d_k^{-1} \end{pmatrix} \quad L_{k-1}^{-1} = \begin{pmatrix} L_{k-1}^{-1} & \\ * & 1 \end{pmatrix}$$

allora $y_k = D_k^{-1} L_k^{-1} e_1 \|b\|_2 = \begin{pmatrix} D_{k-1}^{-1} L_{k-1}^{-1} \hat{e}_1 \|b\|_2 \\ \eta_k \end{pmatrix} = \begin{pmatrix} y_{k-1} \\ \eta_k \end{pmatrix}$ dove $\hat{e}_1$ è il vettore della base canonica in dimensione più bassa. Invece

$$\begin{aligned} \widetilde{P}_k = Q_k L_k^{-T} &= \begin{pmatrix} Q_{k-1} & q_k \end{pmatrix} \begin{pmatrix} L_{k-1}^{-T} & * \\ & 1 \end{pmatrix} \\ &= \begin{pmatrix} Q_{k-1} L_{k-1}^{-T} & \widetilde{p}_k \end{pmatrix} = \begin{pmatrix} \widetilde{P}_{k-1} & \widetilde{p}_k \end{pmatrix} \end{aligned}$$

Osserviamo che $\widetilde{P}_k L_k^T = Q_k$ e in particolare consideriamo la k-esima colonna di entrambe le matrici:

$$\begin{aligned}
(\widetilde{P}_k L_k^T)e_k &= q_k \\
\Rightarrow \begin{pmatrix} \widetilde{P}_{k-1} & \widetilde{p}_k \end{pmatrix} \begin{pmatrix} L_{k-1}^T & l_{k-1}\hat{e}_{k-1} \\ & 1 \end{pmatrix} &= q_k \\
\Rightarrow l_{k-1}\widetilde{P}_{k-1}\hat{e}_{k-1} + \widetilde{p}_k &= q_k \\
\Rightarrow l_{k-1}\widetilde{p}_{k-1} + \widetilde{p}_k &= q_k \\
\Rightarrow \widetilde{p}_k &= q_k - l_{k-1}\widetilde{p}_{k-1}
\end{aligned}$$

Inoltre da

$$\begin{cases} x_k = x_{k-1} + \eta_k \widetilde{p}_k \\ r_k = b - Ax_k \end{cases}$$

segue che $Ax_k = Ax_{k-1} - \eta_k A\widetilde{p}_k \Rightarrow r_k = r_{k-1} - \eta_k A\widetilde{p}_k$ e quindi $p_{k-1} = \|r_{k-1}\|_2 \widetilde{p}_k$ per cui

$$\begin{aligned}
x_k &= x_{k-1} + \frac{\eta_k}{\|r_{k-1}\|_2} p_{k-1} = x_{k-1} + \alpha_{k-1} p_{k-1} \\
r_k &= r_{k-1} - \frac{\eta_k}{\|r_{k-1}\|_2} A p_{k-1} = r_{k-1} - \alpha_{k-1} A p_{k-1}
\end{aligned}$$

e dall'espressione di $\widetilde{p}_k$ trovata sopra si ha

$$\begin{aligned}
p_{k-1} &= \|r_{k-1}\|_2 (q_k - l_{k-1}\widetilde{p}_{k-1}) \\
&= r_{k-1} - l_{k-1} \frac{\|r_{k-1}\|_2}{\|r_{k-2}\|_2} p_{k-2} \\
&= r_{k-1} + \beta_{k-1} p_{k-2}
\end{aligned}$$

ponendo $\alpha_{k-1} := \eta_k / \|r_{k-1}\|_2$ e $\beta_k := -l_{k-1} \frac{\|r_{k-1}\|_2}{\|r_{k-2}\|_2}$.

7.2 Metodo di Arnoldi (FOM)

Detto anche **F**ull **O**rthogonalisation **M**ethod, si usa per risolvere il sistema lineare $Ax = b$. Dato x_0 costruiamo $r_0 = b - Ax_0$ e lo spazio $\mathcal{K}_k(A, r_0) = \text{Span}(\{r_0, Ar_0, \ldots, A^{k-1}r_0\})$. Sia x_k tale che $z_k = x_k - x_0 \in \mathcal{K}_k(A, r_0)$.

Oss. 5: Oss.. $z_k = \phi_{k-1}(A)r_0$ dove ϕ è un polinomio di grado al più $k-1$. Dunque $r_k = r_0 - Az_k = (I - A\phi_{k-1}(A))r_0 = \phi_k(A)r_0$ con ϕ_k polinomio di grado $\leq k$ con $\phi_k(0) = 1$ è detto il polinomio residuo.

Il metodo di Arnoldi costruisce una successione tale per cui $r_k \perp \mathcal{K}_k(A, r_0)$, ovvero imponiamo $v^T(b - Ax_k) = 0 \ \forall v \in \mathcal{K}_k(A, r_0)$. Basta $Q_k^T r_k = 0$ in quanto le colonne di Q_k sono una base (ortonormale) dello spazio di Krylov.

$$\begin{aligned}
Q_k^T r_k &= Q_k^T(b - A(z_k + x_0)) = Q_k^T(r_0 - Az_k) \\
&= Q_k^T r_0 - Q_k^T A z_k = Q_k^T r_0 - Q_k^T A Q_k z \\
&= Q_k^T r_0 - H_k z
\end{aligned}$$

ora risolvendo $H_k z = Q_k^T r_0$ per trovare z posso trovare anche x_k. Tuttavia non ci sono garanzie che H_k sia invertibile e ben condizionata, a meno che A non sia definita positiva e in tal caso lo è anche H_k.

Oss. 6: Oss.. Per come sono fatti i q_i $Q_k^T r_0 = \|r_0\|_2 e_1$ Anche in questo caso, se lavorassimo in aritmetica esatta, avremmo la soluzione in al più n passi. Possiamo dunque considerarlo un metodo iterativo e stabilire un criterio d'arresto. Usando la partizione delle matrici a blocchi

$$\begin{aligned} AQ_k &= Q_k H_k + Q_u H_{ku} \\ &= Q_k H_k + Q_u h_{k+1k} e_1 e_k^T \\ &= Q_k H_k + h_{k+1k} q_{k+1} e_k^T \Rightarrow r_k \quad = r_0 - Q_k H_k z - h_{k+1k} q_{k+1} e_k^T z \\ &= -h_{k+1k} q_{k+1} e_k^T z \\ \Rightarrow \|r_k\|_2 &= |h_{k+1k} e_k^T z| \end{aligned}$$

dunque possiamo sfruttare un criterio d'arresto senza calcolare esplicitamente il residuo.

Oss. 7: Equivalenza. Nel caso A definita positiva, i metodi FOM e CG coincidono.

Oss. 8: Costi e varianti. Questo metodo risulta oneroso sia per occupazione di memoria (ricordare Q_k) che per costo computazionale. Ci sono alcune varianti per migliorarlo:

1. FOM(m) : fissato $m \leq 10$ si fanno al più m passi con il FOM. Se c'è convergenza si termina, altrimenti si ricomincia con $x_0 = x_m$. Per questo è detto anche *FOM con restart*.

2. IOM = **I**ncomplete **O**rthogonalisation **M**ethod che sfrutta una ortogonalizzazione parziale.

7.3 Metodo GMRES

Partendo da $Q_k^T A Q_k = H_k$ ottenuto con l'algoritmo di Arnoldi, si costruisce la successione $x_k = Q_k y_k \in \mathcal{K}_k(A, b)$ in modo da minimizzare

$$\begin{aligned}
\|r_k\|_2 &= \|b - Ax_k\|_2 \\
&= \|b - AQ_k y_k\|_2 \\
&= \|Q(Q^T b - HQ^T Q_k y_k)\|_2 \\
&= \|e_1\|b\|_2 - H \begin{pmatrix} Q_k^T & Q_u^T \end{pmatrix} Q_k y_k\|_2 \\
&= \|e_1\|b\|_2 - H \begin{pmatrix} I_k & 0 \end{pmatrix} y_k\|_2 \\
&= \|e_1\|b\|_2 - \begin{pmatrix} H_k & H_{uk} \\ H_{ku} & H_u \end{pmatrix} \begin{pmatrix} y_k \\ 0 \end{pmatrix}\|_2 \\
&= \|e_1\|b\|_2 - \begin{pmatrix} H_k \\ H_{ku} \end{pmatrix} y_k\|_2
\end{aligned}$$

cioè l'obiettivo è trovare y_k che minimizza e ci siamo ricondotti a un problema di minimi quadrati con matrice strutturata:

$$\begin{pmatrix} H_k \\ H_{ku} \end{pmatrix} = \begin{pmatrix} a & & \\ c & & * \\ 0 & d & b \\ & 0 & \end{pmatrix}$$

[thick]ab [thick]cd che si fattorizza QR utilizzando matrici di rotazione di Givens. Ad ogni passo c'è da risolvere un problema di minimi quadrati. Anche in questo caso c'è la variante con *restart*.

8 L'equazione di Poisson

8.1 Caso in dimensione 1

L'equazione di Poisson è data da

$$-\frac{d^2}{dx^2}v(x) = f(x) \quad \text{per} \quad 0 < x < 1$$

con condizioni (di Dirichlet omogenee) al contorno: $v(0) = v(1) = 0$.
Per risolverla con metodi numerici si può procedere in tre passi:

1. *Discretizzazione* del dominio. Il modo più semplice è scegliere un passo $h = 1/N + 1$ e porre $x_i = ih$ per $i = 0, \dots, N + 1$.

2. *Discretizzazione del problema*: ponendo $v_i = v(x_i)$ ed $f_i = f(x_i)$.

   Oss. 1: Operatori discreti. Se F è sufficientemente regolare si ha:

   $$\begin{aligned} F(x_0 + h) =& F(x_0) + hF'(x_0) + \frac{h^2}{2}F''(x_0) + \frac{h^3}{6}F'''(x_0) + \frac{h^4}{4!}F^{(4)}(\xi_1) \\ F(x_0 - h) =& F(x_0) - hF'(x_0) + \frac{h^2}{2}F''(x_0) - \frac{h^3}{6}F'''(x_0) + \frac{h^4}{4!}F^{(4)}(\xi_2) \\ \Rightarrow F''(x_0) =& \frac{1}{h^2}\left[F(x_0 + h) - 2F(x_0) + F(x_0 + h)\right] + \hat{\tau} \end{aligned}$$

   con $\hat{\tau} = -\frac{h^2}{12}F^{(4)}(\bar{\xi})$ (e quindi $|\tau| \leq \frac{h^2}{12}\max_{x\in[0,1]}|F^{(4)}(x)|$).

3. $\begin{cases} -v_{i-1} + 2v_i - v_{i+1} & = h^2 f_i + h^2\tau_i \;\; 0 < i < N + 1 \\ v_0 & = v_{N+1} = 0 \end{cases}$

   Ora posti

   $$T_N = \begin{pmatrix} 2 & -1 & & \\ -1 & \ddots & \ddots & \\ & \ddots & \ddots & -1 \\ & & -1 & 2 \end{pmatrix} \quad v = \begin{pmatrix} v_1 \\ \vdots \\ v_N \end{pmatrix} \quad f = \begin{pmatrix} f_1 \\ \vdots \\ f_N \end{pmatrix} \quad \tau = \begin{pmatrix} \tau_1 \\ \vdots \\ \tau_N \end{pmatrix}$$

in forma compatta $T_N v = h^2 f + h^2\tau$. trascuriamo per un attimo il termine $h^2\tau$ e otteniamo $T_N\hat{v} = h^2 f$. Abbiamo ricondotto così il problema a risolvere un sistema lineare. T_N ha autovalori $\lambda_j = 2\left(1 - \cos\frac{\pi j}{N+1}\right)$ per $j = 1, \dots, N$ con relativi autovettori (normalizzati in $\|\cdot\|_2$) $z_j = \sin\left(\frac{\pi jk}{N+1}\right)\sqrt{\frac{2}{N+1}}$.
Inoltre se $\Lambda = \begin{pmatrix} \lambda_1 & & \\ & \ddots & \\ & & \lambda_n \end{pmatrix}$ e $Z = \begin{pmatrix} z_1 & \cdots & z_n \end{pmatrix}$ allora $T_N = Z\Lambda Z^T$. Notiamo che $\lambda_1 = 2\left(1 - \cos\frac{\pi}{N+1}\right) \approx 2\left(-\frac{\pi^2}{2(N+1)^2}\right) = \frac{\pi^2}{(N+1)^2}$ e che $\lambda_n =$

$2\left(1-\cos\frac{\pi N}{N+1}\right) \approx 2(1-\cos\pi) = 4$ cioè che gli autovalori sono tra λ_1 e λ_N e $\lim_{N\to+\infty} \lambda_1 = 0$ quindi T_N è definita positiva e dunque il sistema ammette soluzione unica.

$$\begin{aligned} v - \hat{v} =& h^2 T_N^{-1}\tau \\ \|v-\hat{v}\|_2 =& h^2\|T_N^{-1}\|_2\|\tau\|_2 = h^2\frac{(N+1)^2}{\pi^2}\|\tau\|_2 \\ \|v-\hat{v}\|_2 =& \frac{1}{pi^2}\|\tau\|_2 = O(h^2) \end{aligned}$$

Problema di Sturm-Liouville a potenziale nullo

$$\begin{cases} -\frac{d^2}{dx^2}\hat{z}_i(x) = \hat{\lambda}_i z_i(x) \\ \hat{z}_i(0) = \hat{z}_i(1) = 0 \end{cases}$$

ha come soluzione generale l'autofunzione $\hat{z}_i(x) = \alpha\sin(\sqrt{\hat{\lambda}_i}x) + \beta\cos(\sqrt{\hat{\lambda}_i}x)$ e imponendo le condizioni al bordo si ottiene $\beta = 0$, poi imponendo ad esempio $\alpha = 1$ si ha $\sqrt{\hat{\lambda}_i} = i\pi$ e cioè $\hat{\lambda}_i = i^2\pi^2$.

Oss. 2: Oss..T_N è l'operatore discreto che corrisponde a $-\frac{d^2}{dx^2}$ e, per N molto grande, gli autovalori ed autovettori di T_N tendono a $i\pi$ e $\hat{z}_i$.

8.2 L'equazione di Poisson in 2 dimensioni

$$-\frac{\partial^2 v(x,y)}{\partial x^2} - \frac{\partial^2 v(x,y)}{\partial y^2} = f(x,y) \quad \text{per} \quad (x,y) \in \Omega = (0,1)^2$$

con la condizione $v(x,y) = 0$ sul bordo di Ω.

Anche in questo caso consideriamo la griglia di punti equispaziati $x_i = ih$ e $y_j = jh$ con $h = 1/N + 1$ e $i,j = 0,\ldots,N+1$. Poniamo al solito $v_{ij} = v(x_i,y_j)$ e $f_{ij} = f(x_i,y_j)$. Se approssimiamo

$$\begin{aligned} -\frac{\partial^2 v(x,y)}{\partial x^2}\Big|_{x=x_i,y=y_j} &\approx \frac{2v_{ij} - v_{i-1,j} - v_{i+1,j}}{h^2} \\ \frac{\partial^2 v(x,y)}{\partial y^2}\Big|_{x=x_i,y=y_j} &\approx \frac{2v_{ij} - v_{ij-1} - v_{ij+1}}{h^2} \end{aligned}$$

possiamo usare la somma delle approssimazioni come approssimazione della somma e ottenere la cosiddetta *formula a* 5 *punti*:

$$-\frac{\partial^2 v(x,y)}{\partial x^2} - \frac{\partial^2 v(x,y)}{\partial y^2}\Big|_{x=x_i,y=y_j} \approx \frac{4v_{ij} - v_{i-1j} - v_{i+1j} - v_{ij-1} - v_{ij+1}}{h^2}$$

$$v_{0j} = v_{N+1j} = v_{i0} = v_{iN+1} = 0$$

Oss. 3: Oss.. Le incognite sono gli N^2 nodi interni. Ma ci sono N^2 equazioni a disposizione, quindi anche questa volta ci si riconduce a risolvere un sistema lineare la cui soluzione è unica se la matrice è non singolare.

In generale si ha

$$T_{N\times N} = \begin{pmatrix} T_N + 2I_N & -I_N & & \\ -I_N & \ddots & \ddots & \\ & & -I_N & T_N + 2I_N \end{pmatrix}$$

$$T_{N\times N} = \begin{pmatrix} T_N & & \\ & \ddots & \\ & & T_N \end{pmatrix} + \begin{pmatrix} 2I_N & -I_N & & \\ -I_N & \ddots & \ddots & \\ & & -I_N & 2I_N \end{pmatrix}$$

matrice tridiagonale a blocchi che si può riscrivere in termini di prodotto di Kronecker:

$$T_{N\times N} = (I_N \otimes T_N) + (T_N \otimes I_N)$$

Allora $T_{N\times N}$ si può diagonalizzare tramite $Z \otimes Z$ e i suoi autovalori ed autovettori sono

$$\lambda_{ij} = \lambda_i + \lambda_j \qquad z_{ij} = z_i \otimes z_j$$

Oss. 4: Oss.. Il discorso si può generalizzare a n dimensioni sempre sfruttando il prodotto di Kronecker.

Si ha che:

- $(Z\otimes Z)(Z\otimes Z)^T = I$ cioè anche $(Z\otimes Z)$ è ortogonale e la trasformazione di sopra è di similitudine.
- $T_{N\times N} = (Z \otimes Z)(I_N \otimes \Lambda + \Lambda \otimes I_N)(Z \otimes Z)^T$

Se v_{mij} è l'iterata al passo m nel punto (x_i, y_j) del metodo di Jacobi ($x^{(m)} = D^{-1}(B + C)x^{(m-1)} + D^{-1}b$) Allora per $i, j = 1, \ldots, N$

$$v_{m+1ij} = \frac{1}{4}(v_{mi-1j} + v_{mi+1j} + v_{mij-1} + v_{mij+1} + h^2 f_{ij})$$

Oss. 5: Oss..L'aggiornamento si può fare indipendentemente da come sono ordinati i punti della griglia.

Per applicare Gauss-Seidel dobbiamo scegliere un ordinamento diverso da quello naturale (usato prima), perché poco efficiente. Per esempio l'ordinamento *rosso-nero*. I nodi rossi useranno l'informazione vecchia dei nodi neri, mentre i neri useranno l'informazione aggiornata. ($x^{(m)} = (D - B)^{-1}Cx^{(m-1)} + (D - B)^{-1}b = D^{-1}(Bx^{(m)} + cx^{(m-1)} + b)$). Si ottiene così il seguente algoritmo:

```
for all i,j that are RED
    v[m+1][i][j] = (v[m][i-1][j] +v[m][i+1][j] + \\
    v[m][i][j-1] + v[m][i][j-1] +h^2*f[i][j])/4
end for
for all i,j that are BLACK
    v[m+1][i][j] = (v[m+1][i-1][j] +v[m+1][i+1][j] \\
    + v[m+1][i][j-1] + v[m+1][i][j-1] +h^2*f[i][j])/4
end for
```

Anche nel caso di applicazione del metodo SOR si usa l'ordinamento rosso-nero. Il che porta allo stesso algoritmo con la sola modifica che bisogna moltiplicare per ω e aggiungere $(1-\omega)v_{mij}$ (sia per i nodi rossi che per i nodi neri). ($x^{(m)} = (1-\omega)x^{(m-1)} + \omega D^{-1}[Bx^{(m)} + Cx^{(m-1)} + b]$)

8.3 Convergenza

- *Jacobi*:

$$\begin{aligned}
T_{N\times N} =& 4I - (4I - T_{N\times N}) \\
J =& (4I)^{-1}(4I - T_{N\times N}) = I - \frac{1}{4}T_{N\times N} \\
\lambda_{ij} = \lambda_i + \lambda_j =& 2\left(1 - \cos\frac{\pi i}{N+1}\right) + 2\left(1 - \cos\frac{\pi j}{N+1}\right) \\
=& 4 - 2\left(\cos\frac{\pi i}{N+1} + \cos\frac{\pi j}{N+1}\right) \\
\Rightarrow \rho(J) =& \max_{i,j}|1 - \frac{\lambda_{ij}}{4}| = \max_{i,j}|\frac{1}{2}\left(\cos\frac{\pi i}{N+1} + \cos\frac{\pi j}{N+1}\right)| \\
=& |1 - \frac{\lambda_{11}}{4}| = |1 - \frac{\lambda_{NN}}{4}| = \cos\frac{\pi}{N+1} \\
\approx& 1 - \frac{\pi^2}{2(N+1)^2}
\end{aligned}$$

cioè, più N è grande e più il metodo di Jacobi è lento. Inoltre $\lim_{N\to+\infty}\lambda_{ij} = 0$ quindi $T_{N\times N}$ tende ad essere una matrice mal condizionata. Quante iterazioni servono per ridurre l'errore di una quantità e^{-1}? Se m è il numero di iterazioni si deve avere:

$$\begin{aligned}
& \rho(J)^m \approx e^{-1} \iff \left(1 - \frac{\pi^2}{2(N+1)^2}\right)^m \approx e^{-1} \\
\iff\ & m \approx -\frac{1}{\log\left(1 - \frac{\pi^2}{2(N+1)^2}\right)} \approx \frac{2(N+1)^2}{\pi^2} = O(N^2)
\end{aligned}$$

Quindi il numero di iterazioni necessarie è circa $O(n) := O(N^2)$. Il costo per ogni aggiornamento è un $O(1)$ quindi per aggiornare tutte le componenti serve $O(n)$ e il costo dopo n operazioni è un $O(n^2)$.

- *Gauss-Seidel* Si dimostra che $\rho(G) = \rho(J)^2$ associato all'ordinamento rosso-nero, per cui $\rho(G) = \left(\cos\frac{\pi}{N+1}\right)^2 \approx \left(1 - \frac{\pi^2}{2(N+1)^2}\right)^2$. Ne segue che il numero di iterazioni è la metà di quello richiesto per Jacobi. In termini di O grande i costi sono gli stessi.

- *SOR* Scegliendo $1 < \omega_0 = \frac{2}{1+\sin\frac{\pi}{N+1}} < 2$ il parametro ottimale, si ottiene che

$$\rho(H(\omega_0) = \frac{\cos^2\frac{\pi}{N+1}}{\left(1+\sin\frac{\pi}{N+1}\right)^2} \approx 1 - \frac{2\pi}{N+1}$$

  ovvero il metodo SOR è circa N volte più veloce rispetto a Jacobi e Gauss-Seidel. Calcoliamo il numero di iterazioni necessarie:

$$\rho(H(\omega_0))^j \approx e^{-1} \iff \left(1 - \frac{2\pi}{N+1}\right)^j \approx e^{-1}$$
$$\iff j \approx O(N) = O(n^{1/2})$$

  Se dunque $\rho(J)^k \approx e^{-1}$ si ha che

$$\left(1 - \frac{1}{N}\right)^j \approx \left(1 - \frac{1}{N^2}\right)^k$$

  quindi $k \approx jN$. In definitiva il costo totale è (calcolando analogamente a prima) $O(n^{3/2})$.

Oss. 6: Tabella riassuntiva.:

Metodo	**Tempo**	**Spazio**
Cholesky	n^3	n^2
Jacobi	n^2	n
Gauss-Seidel	n^2	n
CG	$n^{3/2}$	n
SOR	$n^{3/2}$	n

tenendo conto che Cholesky è un metodo diretto, quindi esatto, mentre i metodi iterativi sono approssimati e che tutto dipende dall'implementazione. (Si è posto sempre $n := N^2$).

9 Metodi del Gradiente

Per il problema della risoluzione di sistemi lineari di grosse dimensioni con struttura. Supponiamo $A \in \mathbb{R}^{n\times n}$ definita positiva, $b \in \mathbb{R}^n$ e di avere il sistema $Ax = b$. Riconduciamo il problema alla ricerca del minimo del funzionale

$$\Phi(x) = \frac{1}{2}x^T Ax - b^T x$$

chiamando *residuo*

$$r(x) := b - Ax = -\nabla\Phi(x) = \left[\begin{array}{ccc} -\frac{\partial\Phi}{\partial x_1} & \cdots & -\frac{\partial\Phi}{\partial x_n} \end{array} \right]^T$$

Oss. 1: Oss.. $\nabla\Phi(x) = 0 \iff r(x) = 0 \iff Ax = b$ e l'hessiano di $\Phi(x)$ è $A > 0$ quindi x è punto di minimo per il funzionale Φ:

$$Ax^* = b \iff x^* = \min_{x\in\mathbb{R}^n} \Phi(x)$$

Sia x_k l'approssimazione di x^* al passo k. La successione si definisce come segue:

- si prende un vettore $p_k \neq 0$ che sia direzione di decrescita per Φ, cioè tale che $p_k^T \nabla\Phi(x_k) < 0$.
- preso α_k tale che $\Phi(x_k) = \min_{\alpha\in\mathbb{R}} \Phi(x_k + \alpha p_k)$ si pone

$$x_{k+1} = x_k + \alpha p_k$$

e si nota che

$$\begin{aligned} \frac{\partial\Phi}{\partial\alpha} = 0 &\iff (x^T + \alpha p_k)^T A p_k - b^T p_k = 0 \\ &\iff \alpha_k = \frac{(b - Ax_k)^T p_k}{p_k^T A p_k} \end{aligned}$$

e posto $r_k := r(x_k)$ l'espressione diviene $\alpha_k = \frac{r_k^T p_k}{p_k^T A p_k}$ e si ha $p_k^T r_k > 0$ e quindi $\alpha_k > 0$.

Oss. 2: Oss..

$$r_{k+1} = b - Ax_{k+1} = b - A(x_k + \alpha_k p_k) = r_k - \alpha_k A p_k$$

e anche

$$r_{k+1}^T p_k = (r_k - \alpha_k A p_k)^T p_k = r_k^T p_k - \alpha_k p_k^T A p_k = 0$$

9.1 Steepest descent

La scelta $p_k = r_k = -\nabla\Phi(x_k)$ dà origine al metodo *steepest descent*. In questo caso $r_{k+1}^T r_k = 0$ e cioè $p_{k+1}^T p_k = 0$. Stimiamo l'errore $e_k = x^* - x_k$: detti λ_{min} e λ_{max} gli autovalori di A di modulo minimo e massimo rispettivamente si ottiene:

$$e_{k+1}^T A e_{k+1} \leq \left(\frac{\lambda_{max} - \lambda_{min}}{\lambda_{max} + \lambda_{min}}\right)^2 e_k^T A e_k$$

da cui posta per definizione $\|x\|_A := \sqrt{x^T A x}$

$$\|e_{k+1}\|_A \leq \left(\frac{\lambda_{max} - \lambda_{min}}{\lambda_{max} + \lambda_{min}}\right) \|e_k\|_A$$

$$\Rightarrow \|e_{k+1}\|_A \leq \left(\frac{\mu_2(A) - 1}{\mu_2(A) + 1}\right)^{k+1} \|e_0\|_A$$

dove $\mu_2(A) = \frac{\lambda_{max}}{\lambda_{min}}$ quindi l'errore si abbatte velocemente se A è ben condizionata.

9.2 Metodo del gradiente coniugato

Scegliamo

$$p_k = \begin{cases} r_0 & \text{se } k = 0 \\ r_k + \beta_k p_{k-1} & k \geq 1 \end{cases}$$

con β_k tale che $p_k^T A p_{k-1} = 0$ cioè tale che le direzioni p_k e p_{k-1} sono A-coniugate. Cerchiamo un espressione per β_k:

$$\begin{aligned} p_k^T A p_{k-1} = (r_k + \beta_k p_{k-1})^T A p_{k-1} &= 0 \\ \Rightarrow r_k^T A p_{k-1} + \beta_k p_{k-1}^T A p_{k-1} &= 0 \\ \Rightarrow \beta_k = -\frac{r_k^T A p_{k-1}}{p_{k-1}^T A p_{k-1}} & \end{aligned}$$

e verifichiamo che p_k così definita è una direzione di decrescita:

$$\begin{aligned} p_k^T \nabla\Phi(x_k) &= -p_k^T r_k \\ &= -(r_k + \beta_k p_{k-1})^T r_k \\ &= -r_k^T r_k - \beta_k p_{k-1}^T r_k \\ &= -\|r_k\|_2^2 < 0 \end{aligned}$$

Da questo segue anche che $p_k^T r_k = r_k^T r_k$ e quindi otteniamo un'espressione per $\alpha_k = \frac{r_k^T p_k}{p_k^T A p_k} = \frac{r_k^T r_k}{p_k^T A p_k}$. Osserviamo inoltre che due residui consecutivi sono ortogonali: $r_k^T r_{k-1} = 0$. Sfruttando le due espressioni per $p_k^T r_{k-1}$ ottenute sostituendo prima la definizione di p_k e poi quella di r_{k-1}, si ottiene una espressione alternativa per $\beta_k = \frac{r_k^T r_k}{r_{k-1}^T r_{k-1}}$. Con queste manipolazioni il costo per il calcolo di α_k e β_k è quello dei tre prodotti scalari.

Teorema 9.1. *Sia $r_0 \neq 0$ e $h \geq 1$ per cui $\forall k \leq h\ r_k \neq 0$. Allora*

$$r_k^T r_j = 0 \quad p_k^T A p_j = 0 \quad k \neq j \;\; k, j = 0, \ldots, h$$

cioè i primi h resisdui sono un insieme di vettori ortogonali e i p_k sono A-coniugati con tutti gli altri.

Dimostrazione. Mostriamo il risultato per induzione. Se $h = 1$ allora $k = 1, j = 0$ e dunque $r_1^T r_0 = 0$ e $p_1^T A p_0 = 0$ perché consecutivi. Il passo induttivo consiste nel provare che per $j = 0, \ldots, h$ si ha:

$$\begin{cases} r_{h+1}^T r_j = 0 \\ p_{h+1}^T A p_j = 0 \end{cases}$$

ed il caso $j = h$ è ovvio come sopra. Abbiamo allora:

$$\begin{aligned} r_{h+1}^T r_j =& (r_h - \alpha_h A p_h)^T r_j \\ =& r_h^T r_j - \alpha_h p_h^T A r_j \\ =& -\alpha_h p_h^T A r_j \qquad \text{per ipotesi induttiva} \\ =& -\alpha_h p_h^T A (p_j - \beta_k p_{j-1}) \\ =& -\alpha_h p_h^T A p_j + \alpha_h \beta_k p_h^T A p_{j-1} \\ =& 0 \end{aligned}$$

dove l'ultima uguaglianza segue ancora dall'ipotesi induttiva. Inoltre:

$$\begin{aligned} p_{h+1}^T A p_j =& (r_{h+1} + \beta_{h+1} p_h)^T A p_j \\ =& r_{h+1}^T A p_j + \beta_{h+1} p_h^T A p_j \\ =& r_{h+1}^T A p_j \\ =& \frac{1}{\alpha_j} r_{h+1}^T (r_j - r_{j+1}) \\ =& \frac{1}{\alpha_j} (r_{h+1}^T r_j - r_{h+1}^T r_{j+1}) \\ =& 0 \end{aligned}$$

dove abbiamo utilizzato due volte l'ipotesi induttiva e il fatto che $A p_j = \frac{1}{\alpha_j}(r_j - r_{j+1})$. □

Oss. 3: Criterio d'arresto. Se lavorassimo in aritmetica esatta non ci potrebbero essere più di n vettori ortogonali in uno spazio di dimensione n. Quindi significherebbe che $\exists m \leq n\ r_m = 0$ e che la soluzione esatta si trova in un numero finito di passi. Lavorando in aritmetica finita, il metodo del

gradiente si comporta come un metodo iterativo. Allora bisogna determinare un criterio d'arresto. Si può usare il criterio del residuo: $\|r_k\|_2 \leq \varepsilon\|b\|_2$.

Oss. 4: Algoritmo del CG.

$$
\begin{aligned}
&k = 0; \quad x_0 = 0; \quad r_0 = b;\\
&p_0 = r_0; \quad \nu_0 = r_0^T r_0; \quad \text{tol} = \varepsilon\nu_0;\\
&\qquad \textbf{while}(\nu_k \geq tol)\\
&\qquad\qquad w = Ap_k;\\
&\qquad\qquad \alpha_k = v_k/(p_k^T w);\\
&\qquad\qquad x_{k+1} = x_k + \alpha_k p_k\\
&\qquad\qquad r_{k+1} = r_k - \alpha_k w\\
&\qquad\qquad \nu_{k+1} = r_{k+1}^T r_{k+1}\\
&\qquad\qquad \beta_{k+1} = \nu_{k+1}/\nu_k\\
&\qquad\qquad p_{k+1} = r_{k+1} + \beta_{k+1} p_k\\
&\qquad\qquad k = k + 1
\end{aligned}
$$

Si dimostra che $\|e_k\|_A \leq 2\left(\frac{\sqrt{\mu_2(A)}-1}{\sqrt{\mu_2(A)}+1}\right)^k \|e_0\|_A$.

Oss. 5: Mal condizionamento. Se la matrice A è mal condizionata si possono applicare delle tecniche per ovviare al problema.

Definizione 9.1. Una matrice $M = C_1C_2$ è un **precondizionatore** (per la matrice A) se C_1, C_2 sono ivertibili e tali che $\mu_2(C_1^{-1}AC_2^{-1} << \mu_2(A)$. Nei casi specifici in cui sia $C_1 = I$ o $C_2 = I$ si parla di precondizionamento rispettivamente destro o sinistro.

Se $y = C_2x$ il sistema $Ax = b$ diventa

$$C_1^{-1}AC_2^{-1}C_2x = C_1^{-1}b \iff (C_1^{-1}AC_2^{-1})y = C_1^{-1}b$$

per cui basta risolvere $C_2x = y$. Essendo A definita positiva, scegliamo $M = CC^T$ in modo che

$$Ax = b \iff (C^{-1}AC^{-T})C^Tx = C^{-1}b \iff By = c$$

Altre scelte possibili per M

- $M =$diag$(a_{11}, \ldots, a_{nn})$
- $M = \begin{pmatrix} \boxed{A_{11}} & & \\ & \ddots & \\ & & \boxed{A_{nn}} \end{pmatrix}$ diagonale a blocchi con i blocchi corrispondenti in A.

- $M = LL^T$ fattorizzazione incompleta di Cholesky (imponendo che $a_{ij} = 0 \Rightarrow l_{ij} = 0$).

La scelta del precondizionatore dipende da caso a caso a seconda della matrice A.

Oss. 6: Oss.. Per come abbiamo scelto M, anche la matrice B è definita positiva.

Applichiamo il metodo del gradiente coniugato al nuovo sistema:

$$\begin{aligned} s_k := r(y_k) &= c - By_k \\ &= C^{-1}b - C^{-1}AC^{-T}C^T x_k \\ &= C^{-1}(b - Ax_k) \\ &= C^{-1}r_k \Rightarrow s_k^T s_K = \qquad\qquad r_k^T C^{-T} C^{-1} r_k \\ &= r_k^T M^{-1} r_k \end{aligned}$$

Se definiamo z_k come soluzione di $Mz_k = r_k$ otteniamo che $s_k^T s_k = r_k^T z_k$ e si ha l'algoritmo del gradiente coniugato precondizionato:

Oss. 7: Algoritmo del PCG.

$k = 0; \quad x_0 = 0; \quad r_0 = b;$
calcolo z_0 t.c. $Mz_0 = r_0$
$p_0 = C^T z_0; \quad \nu_0 = z_0^T r_0; \quad \text{tol} = \varepsilon\nu_0;$
while$(\nu_k \geq tol)$
$w = Ap_k;$
$\alpha_k = v_k/(p_k^T w);$
$x_{k+1} = x_k + \alpha_k p_k$
$r_{k+1} = r_k - \alpha_k w$
z_{k+1} t.c. $Mz_{k+1} = r_{k+1}$
$\nu_{k+1} = r_{k+1}^T z_{k+1}$
$\beta_{k+1} = \nu_{k+1}/\nu_k$
$p_{k+1} = C^T z_{k+1} + \beta_{k+1} p_k$
$k = k + 1$

9.3 CG e minimi quadrati

Possiamo usare il metodo del gradiente coniugato per risolvere il problema dei minimi quadrati.

Date come sempre $A \in \mathbb{R}^{m\times n},\ b \in \mathbb{R}^m\ m \geq n\ Ax = b$ dove cerchiamo x che minimizza $\min_{y\in\mathbb{R}^n} \|Ay - b\|_2^2$.

$$\|Ay - b\|_2^2 = (Ay - b)^T(Ay - b) = 2(\frac{1}{2}y^T A^T Ay - b^T Ay) + b^T b$$

quindi il funzionale da minimizzare è

$$\Psi(x) = \min_{y\in\mathbb{R}^n} \Psi(y) = \min_{y\in\mathbb{R}^n} \frac{1}{2}(\|Ay - b\|_2^2 - b^T b)$$

il cui gradiente è:

$$\begin{aligned} -\nabla\Psi(y) &= A^T(b - Ay) = A^T r(y) \\ \Rightarrow -\nabla\Psi(x_k) &= A^T r_k \end{aligned}$$

Oss. 8: Oss.. Rispetto al funzionale $\Phi(x)$ la matrice è diventata $A^T A$.

Definiamo $s_k := A^T r_k$ e costruiamo la successione $x_{k+1} = x_k + \alpha_k p_k$ con α_k tale che $\frac{\partial\Psi}{\partial\alpha} = 0$ e cioè $\alpha_k = \frac{p_k^T s_k}{\|Ap_k\|_2^2}$. Moltiplicando per A a sinistra si ottiene $r_{k+1} = r_k - \alpha_k Ap_k$ e $p_k = \begin{cases} s_0 = A^T r_0 & k = 0 \\ s_k + \beta_k p_{k-1} & k \geq 1 \end{cases}$ con $\beta_k = -\frac{s_k^T A^T Ap_{k-1}}{\|Ap_{k-1}\|_2^2}$ in modo tale che $p_k^T A^T Ap_{k-1} = 0$.

Verifichiamo che p_k è una direzione di discesa:

$$\begin{aligned} -p_k^T\nabla\Psi(x_k) &= p_k^T A^T r_k = p_k^T s_k \\ &= (s_k + \beta_k p_{k-1})^T s_k \\ &= s_k^T s_k + \beta_k p_{k-1}^T s_k \\ &= s_k^T s_k \geq 0 \end{aligned}$$

notando che $p_{k-1}^T s_k = p_{k-1}^T A^T r_k = p_{k-1}^T A^T r_{k-1} - \alpha_k p_{k-1}^T A^T Ap_{k-1} = p_{k-1}^T s_{k-1} - p_k^T s_{k-1} = 0$ per definizione di α_k ed s_k. Dalla relazione precedente segue anche che $p_k^T s_k = \|s_k\|_2^2$ e si può quindi riscrivere $\alpha_k = \frac{\|s_k\|_2^2}{\|Ap_k\|_2^2}$ e $\beta_k = \frac{\|s_k\|_2^2}{\|s_{k-1}\|_2^2}$.

Esplicitiamo il calcolo di β_k usando la relazione tra i residui:

$$\begin{aligned} \beta_k &= \frac{-s_k^T A^T Ap_{k-1}}{\|s_{k-1}\|_2^2}\alpha_{k-1} \\ &= \frac{s_k^T A^T Ap_{k-1}}{\|s_{k-1}\|_2^2}\frac{(r_k - r_{k-1})}{Ap_{k-1}} \\ &= \frac{s^T A^T r_k}{\|s_{k-1}\|_2^2} - \frac{s_k^T A^T r_{k-1}}{\|s_{k-1}\|_2^2} \\ &= \frac{s_k^T s_k}{\|s_{k-1}\|_2^2} - \frac{s_k^T s_{k-1}}{\|s_{k-1}\|_2^2} \\ &= \frac{\|s_k\|_2^2}{\|s_{k-1}\|_2^2} \end{aligned}$$

infatti

$$\begin{aligned} s_k^T s_{k-1} &= (r_k^T A) s_{k-1} \\ &= (r_{k-1} - \alpha_{k-1} A p_{k-1})^T A A^T r_{k-1} \\ &= r_{k-1}^T A A^T r_{k-1} - \alpha_{k-1} p_{k-1}^T A^T A A^T r_{k-1} \\ &= \|s_{k-1}\|_2^2 - \alpha_{k-1} (A p_{k-1})^T A s_{k-1} \\ &= \|s_{k-1}\|_2^2 - \alpha_{k-1} (A p_{k-1})^T A (p_{k-1} - \beta_{k-1} p_{k-2}) \\ &= \|s_{k-1}\|_2^2 + \alpha_{k-1} \beta_{k-1} p_{k-1}^T A^T A P_{k-2} - \alpha_{k-1} \|A p_{k-1}\|_2^2 \\ &= 0 \end{aligned}$$

per costruzione di α_{k-1} e direzioni $A^T A$-coniugate.

Infine si ottiene l'algoritmo:

Oss. 9: Minimi quadrati con CG.

$$\begin{aligned} k = 0; \quad x_0 = 0; \quad r_0 = b; \quad s_0 = A^T r_0; \\ p_0 = s_0; \quad \nu_0 = s_0^T s_0; \quad \text{tol} = \varepsilon \nu_0; \\ \textbf{while}(\nu_k \geq tol) \\ w = A p_k; \\ \alpha_k = v_k / (w^T w); \\ x_{k+1} = x_k + \alpha_k p_k \\ r_{k+1} = r_k - \alpha_k w \\ s_{k+1} = A^T r_{k+1} \\ \nu_{k+1} = s_{k+1}^T s_{k+1} \\ \beta_{k+1} = \nu_{k+1} / \nu_k \\ p_{k+1} = s_{k+1} + \beta_{k+1} p_k \\ k = k + 1 \end{aligned}$$

Se $r(x) = b - Ax$ e x^* è soluzione esatta di $Ax = b$, allora

$$\|x^* - x_k\| = \|A^{-1} b - x_k\| = \|A^{-1}(b - A x_k)\| = \|A^{-1} r_k\|$$

$$\text{e} \frac{\|x^* - x_k\|}{x^* - x_0} \leq \frac{\|A^{-1}\| \|r_k\|}{\frac{\|r_0\|}{\|r_k\|}} = \mu(A) \frac{\|r_k\|}{\|r_0\|}$$

Oss. 10: Oss.. Se $x_0 = 0, \|r_0\| = \|b\|$, cioè se la matrice A è ben condizionata, il criterio di arresto del residuo ci permette di controllare bene l'errore relativo. In generale, così sappiamo come diminuisce l'errore ad ogni passo.

A Appendice

Qui di seguito risultati non provati o enunciati durante il corso di calcolo scientifico, ma che sono correlati.

Proposizione A.1 (Positività del complemento di Schur.)**.** *Sia A una matrice simmetrica e definita positiva. Sia quindi*

$$M = \begin{pmatrix} A & B \\ B^T & C \end{pmatrix}$$

con C simmetrica. Allora $M \geq 0 \iff C - B^T A^{-1} B \geq 0$.

Dimostrazione. Per provare la proposizione basta prendere un vettore $x = \begin{pmatrix} u \\ v \end{pmatrix}$ e applicare la definizione:

$$x^T M x \geq 0 \iff u^T A u + 2 v^T B^T u + v^T C v \geq$$

e minimizzando $u \mapsto u^T A u + 2 v^T B^T u + v^T C v$ per v fissato, otteniamo $u = -A^{-1} B^T v$ che sostituendo nella funzione per trovare il valore minimo dà il risultato voluto. □

A.1 Invertire matrici

Una collezione di formule per invertire la somma di matrici:

Woodbury matrix identity: A, C invertibile:

$$(A+UCV)^{-1} = A^{-1} - A^{-1}U\left(C^{-1}+VA^{-1}U\right)^{-1}VA^{-1} \tag{1a}$$

when $U = V = I$

$$(A+C)^{-1} = A^{-1} - A^{-1}\left(C^{-1}+A^{-1}\right)^{-1}A^{-1} \tag{1b}$$

when $C = V = I$

$$(A+U)^{-1} = A^{-1} - A^{-1}U(A+U)^{-1} \tag{1c}$$

Sherman–Morrison formula: A invertibile:

$$(A+uv^T)^{-1} = A^{-1} - \frac{A^{-1}uv^TA^{-1}}{1+v^TA^{-1}u} \tag{1d}$$

No requirements on P or Q :

$$(I+P)^{-1} = I - (I+P)^{-1}P \tag{1e}$$

$$(I+PQ)^{-1}P = P(I+QP)^{-1} \tag{1f}$$

If A is invertible:

$$(A+BCD)^{-1} = A^{-1} - (I+A^{-1}BCD)^{-1}A^{-1}BCDA^{-1} \tag{1g}$$

$$= A^{-1} - A^{-1}(I+BCDA^{-1})^{-1}BCDA^{-1} \tag{1h}$$

$$= A^{-1} - A^{-1}B(I+CDA^{-1}B)^{-1}CDA^{-1} \tag{1i}$$

$$= A^{-1} - A^{-1}BC(I+DA^{-1}BC)^{-1}DA^{-1} \tag{1j}$$

$$= A^{-1} - A^{-1}BCD(I+A^{-1}BCD)^{-1}A^{-1} \tag{1k}$$

$$= A^{-1} - A^{-1}BCDA^{-1}(I+BCDA^{-1})^{-1} \tag{1l}$$

Se C è invertibile ritroviamo la formula di Woodbury (1a):

$$(A+BCD)^{-1} = A^{-1} - A^{-1}B(C^{-1}+DA^{-1}B)^{-1}DA^{-1}$$

A.1.1 Stime sulle norme matriciali

È facile vedere che, se $\|A\| < 1$ per una qualche norma di matrice indotta $\|\cdot\|$, allora $I - A$ è invertibile e, da (1e),

$$\|(I-A)^{-1}\| \leq \frac{1}{1-\|A\|}.$$

Possiamo generalizzare come segue: supponiamo $\|A\| < m$. Allora $I - \frac{1}{m}A$ è invertibile e

$$\|\left(I-\frac{1}{m}A\right)^{-1}\| \leq \frac{1}{1-\frac{\|A\|}{m}} = \frac{m}{m-\|A\|}$$

e possiamo riscrivere

$$\|(mI-A)^{-1}\| \leq \frac{1}{m-\|A\|}.$$

Da questa stima si può derivare una minorazione per la matrice inversa:

$$1 + \|A\| \geq \|I - A\| \geq \frac{1}{\|(I-A)^{-1}\|} \geq 1 - \|A\|.$$

Proposizione A.2 (Norme matriciali e raggio spettrale)**.** *Per ogni matrice A ed $\varepsilon > 0$ esiste una norma matriciale indotta tale che, se $\rho(A)$ è il raggio spettrale,*

$$\rho(A) \leq \|A\| \leq \rho(A) + \varepsilon$$

Inoltre, se tutti gli autovalori μ di A tali che $|\mu| = \rho(A)$ hanno corrispondenti blocchi di Jordan di dimensione 1*, allora esiste una norma matriciale indotta per cui $\|A\| = \rho(A)$.*

Osservazione 1. Per ogni norma matriciale indotta, se $\rho(A)$ è il raggio spettrale di una matrice A,

$$\rho(A) \leq \|A\|.$$

In fatti se x è un autovettore di A con autovalore λ:

$$\|A\|\|x\| \geq \|Ax\| = \|\lambda x\| = |\lambda|\|x\|$$

il che implica $\|A\| \geq |\lambda|$ per ogni autovalore λ e quindi $\|A\| \geq \rho(A)$.

List of Theorems

List of Definitions

List of Lemmas