Laboratorio di Algoritmi Numerici di interesse per l'Ingegneria

Garau Andrea

Matematica Computazionale

Laboratorio di Algoritmi Numerici di interesse per l'Ingegneria

Garau Andrea
47017

Italia • Sardegna • Cagliari • via Ospedale

Prof. *Giuseppe Rodriguez*
Studente: *Garau Andrea*

Stampato in Sardegna, Assemini

Questo documento è stato composto dall'autore mediante LaTeX, su un sitema GNU OS: Arch Linux $x86\_64$ con Kernel Release $4.10.2 - 1 - ARCH$. Per la stesione del documento, così come per il codice prodotto, si è usato il programma `free software` **Emacs**. L'unico software commerciale utilizzato, per i soli esperimenti numerici, è Matlab®. I grafici sono stati prodotti con il pacchetto TikZ.

Per maggiori informazioni in merito garau.and@gmail.com

Introduzione

L'OBIETTIVO di questo elaborato è quello di analizzare alcuni algoritmi numerici, e simularli praticamente. Questo lavoro si basa principalmente sulle lezioni del corso di *Matematica Computazionale* tenuto dal Prof.Rodriguez a.a. 2016/2017[1]. Non c'è nulla che la matematica non possa descrivere. Conoscere il **modello matematico** del fenomeno, un complesso di formule che descrivono il suo comportamento, vuol dire conoscere **tutto**[2] del fenomeno. Talvolta, diciamo pure il 99.99...9% della volte, le equazioni così ottenute sono troppo complesse per essere risolte, si associa allora un **problema numerico** allo scopo di renderlo risolubile numericamente su un calcolatore, attraverso una sequenza: **l'algoritmo**.

Definizione 0.0.1: Algoritmo

Una sequenza univoca di un numero finito di operazioni elementari che stabilisce come calcolare la soluzione di un problema, assegnati certi dati iniziali.

Un concetto **fondamentale** è quello di **problema ben posto**. J.Hadamard né diede una definizione.

Definizione 0.0.2: Problema ben posto

Un problema è **ben posto** se esso possiede, in un prefissato **campo di definizione**, **una e una sola** soluzione e questa dipende con **continuità** dai dati: In caso contrario, viene detto mal posto.

Nota storica 0.0.1: J.Hadamard

Jacques Solomon Hadamard (Versailles, 8 dicembre 1865 – Parigi, 17 ottobre 1963) è stato un matematico francese, conosciuto principalmente per la sua dimostrazione del teorema dei numeri primi. [Wiki16c].

[1] Ovviamente solo quelle che ha seguito l'autore!
[2] Occorrerebbe dare la rigorosa definizione di tutto.

Organizzazione del lavoro

Capitolo 1. Nel primo capitolo si esamineranno gli errori inevitabilmente presenti quando si fanno misure e calcoli. Come vengono memorizzati i numeri reali su un calcolatore e quali sono le principali cause di origine e propagazione degli errori.

Capitolo 2. In questo secondo capitolo si richiamano alcuni concetti di Algebra lineare, delle nozioni che verranno utilizzate nel seguito del lavoro.

Capitolo 3. La fattorizzazione QR.

Capitolo 4 – 5 – 6. Sono rispettivamente: i metodi diretti, autovalori e autovettori e i SVD. In ogni capitolo si svolge la trattazione teorica[3] Nel 4 in particolar modo è presente l'Algoritmo di Cholesky.

Capitolo 7. Metodi iterativi, spazi di Krylov, Arnoldi Lanczos e GMRES. Non viene trattata[4] la parte del metodi del gradiente.

Capitolo 8. La presentazione sottoforma di grafici di alcuni Algoritmi simulati e confrontati con quelli di Matlab® .

Nel testo sono *sparse* delle *Note storiche*[5] prima di iniziare si è ritenuto doveroso riportare una Nota sul *Princeps mathematicorum* [6].

Nota storica 0.0.2: Johann Friedrich Carl Gauss

Johann Friedrich Carl Gauss (Braunschweig, 30 aprile 1777 – Gottinga, 23 febbraio 1855) è stato un matematico, astronomo e fisico tedesco, che ha dato contributi determinanti in analisi matematica, teoria dei numeri, statistica, calcolo numerico, geometria differenziale, geodesia, geofisica, magnetismo, elettrostatica, astronomia e ottica.
Talvolta definito *il Principe dei matematici* (Princeps mathematicorum) come Eulero o *il più grande matematico della modernità* (in opposizione ad Archimede, considerato dallo stesso Gauss come il maggiore fra i matematici dell'*antichità*), è annoverato fra i più importanti matematici della storia avendo contribuito in modo decisivo all'evoluzione delle scienze matematiche, fisiche e naturali. Definì la matematica come *la regina delle scienze*. [Wiki17d].

[3] Sono gli appunti presi a lezione.
[4] Anche se stata svolta.
[5] Come quella di sopra di J.Hadamard.
[6] Dove il suo nome, anche indirettamente, appare in molti testi di matematica.

Indice

Elenco delle figure

Elenco delle tabelle

Capitolo 1

Errori

La presenza di errori nei calcoli può essere dovuta a varie cause:

- il problema in studio potrebbe essere affetto da errori sperimentali
- un errata modellizzazione del fenomeno
- il problema deve essere posto in una forma numerica più semplice per potere essere realizzabile
- la memorizzazione dei dati su un calcolatore digitale introduce errori di arrotondamento.

Occorre poter quantificare l'errore, sia a la quantità da stimare e a^* la sua approssimazione, definiamo le quantità:

Definizione 1.0.1: Errore assoluto

$$\epsilon = |a - a^*|$$

di difficile interpretazione se non si hanno informazioni sull'ordine di grandezza delle quantità da stimare.

Definizione 1.0.2: Errore relativo

$$\rho = \frac{|a - a^*|}{|a|}$$

1.1 Numeri macchina

I numeri di macchina sono quei numeri che possono essere rappresentati esattamente su un calcolatore, essi includono:

1. i numeri interi, non troppo grandi

2. i numeri con parte decimale, che possono essere rappresentati mediante due differenti tecniche:
   - numeri a virgola fissa riserva t cifre per la parte intera, s per la parte decimale:
     $$(\alpha_{t-1}\alpha_{t-2}\dots\alpha_1\alpha_0.\alpha_{-1}\alpha_{-2}\dots\alpha_{-s})_\beta$$
     non possono essere espressi numeri maggiori o uguali a β^t o minori di β^{-s}
   - numeri a virgola mobile, si memorizzano separatamente le cifre significative e l'ordine di grandezza del numero:
     $$(0.\alpha_1\alpha_2\dots\alpha_t)_\beta\beta^p$$
3. per i numeri complessi si memorizzano separatamente parte reale e immaginaria separatamente, in due variabili reali.

1.1.1 Insieme dei numeri macchina

L'insieme dei numeri macchina in base β, con t cifre significative ed esponente nell'intervallo $[L,U]$ è l'insieme:

$$\mathbb{F}(\beta,t,L,U) = \{\,0\,\} \bigcup \{\, x \in \mathbb{R} : x = sign(x)\cdot m\cdot\beta^p \,\}$$

dove

- t e β son interi positivi, con $\beta \geq 2$;
- $sign(x) = \pm 1$ a seconda del segno di x;
- la quantità:
  $$m = \sum_{i=1}^{t} d_i\beta^{-i} = (0.d_1d_2\dots d_t)_\beta$$
  è la mantissa $(0 \leq d_i \leq \beta - 1)$;
- p è un intero compreso tra L e U, detto esponente o caratteristica;
- $d_1 \neq 0$ normalizzazione, non sempre viene applicata, il quel caso si parla di virgola mobile normalizzata.

L'insieme dei numeri macchina $\mathbb{F}$ è un sottoinsieme proprio di $\mathbb{R}$[1], dato che l'insieme è discreto e finito, **non** può contenere tutti i numeri reali. Ridefiniamo la funzione

$$\begin{aligned} fl : \mathbb{R} &\longrightarrow \mathbb{F} \\ x &\longmapsto fl(x) \end{aligned}$$

che ad ogni numero reale fà corrispondere un rappresentante in $\mathbb{F}$. Tale funzione è suriettiva[2], ma non iniettiva[3]. Dato $x \in \mathbb{R}$ possiamo avere:

[1] Dati due insiemi A e B non vuoti, diremo che A è un sottoinsieme proprio di B se tutti gli elementi dell'insieme A appartengono anche all'insieme B e almeno un elemento dell'insieme B non appartiene all'insieme A: $A \subset B$ cioè è valida l'inclusione stretta.

[2] Una funzione suriettiva quando ogni elemento del codominio è immagine di **almeno** un elemento del dominio.

[3] Una funzione iniettiva è una funzione che associa ad elementi distinti del dominio, elementi distinti del codominio.

1. $x \in \mathbb{F}$ Questo vuol dire che l'esponente è nell'intervallo $[L, U]$ e la mantissa no ha più di t cifre significative se rappresentata in base β.

2. $|x| < \beta^{L-1}$ L'esponente p è minore di L. Un numero troppo piccolo che genera **underflow**.

3. $|x| \geq \beta^{U}$ L'esponente p è maggiore di U. Questa situazione viene detta di **overflow**.

4. $|x| \in [\beta^{L-1}, \beta^{U}), x \notin \mathbb{F}$ In questo caso il numero di cifre significative di x è superiore a t: occorre scegliere una rappresentazione che approssimi x con sufficiente accuratezza, sono possibili alcune tecniche:

   Troncamento le cifre in eccesso vengono trascurate

   Arrotondamento questo procedimento equivale ad aggiungere $\frac{1}{2}\beta^{-t}$ alla mantissa di x e successivamente troncare il risultato.

Ha interesse considerare la **precisione macchina**:

Definizione 1.1.1: Precisione macchina

la distanza tra 1 e il successivo numero di macchina.

$$\epsilon_M = \beta^{1-t}$$

1.1.2 Operazioni

Le normali operazioni matematiche $+, -, *, /$ nel calcolatore diventano le **operazioni macchina**: $\oplus, \ominus, \otimes, \oslash$. L'errore relativo commesso da un'operazione di macchina deve quindi essere dello stesso ordine di grandezza di quello di arrotondamento commesso nel memorizzare il risultato esatto. Quando un sistema in virgola mobile non verifica questa ipotesi si parla di: **aritmetica aberrante**.

Capitolo 2

Richiami di Algebra Lineare

In questo capitolo si concentrano dei risultati importanti utili successivamente nel testo. Si darà prima la definizione di spazio lineare e le sue proprietà, si vedranno gli spazi normati e quelli di Hilbert. Si analizzeranno le matrici e alcune proprietà importanti s esse. Verranno anche studiati gli autovalori e autovettori, e si vedranno alcune matrici che possiedono una particolare struttura. Infine l enorme matriciali e il numero di condizionamento, di relativa importanza per la soluzione di sistemi lineari.

2.1 Spazi

Definizione 2.1.1: Spazio Lineare

Uno spazio lineare o vettoriale reale è un insieme V su cui sono definite due operazioni di somma e prodotto per uno scalare

$$+ : V \times V \longrightarrow V \qquad \cdot : \mathbb{R} \times V \longrightarrow V$$
$$(\mathtt{x}, \mathtt{y}) \longmapsto \mathtt{x} + \mathtt{y} \qquad (\alpha, \mathtt{x}) \longmapsto \alpha\mathtt{x}$$

che, per ogni $\alpha, \beta \in \mathbb{R}$ e $\mathtt{x}, \mathtt{y}, \mathtt{z} \in V$ godono delle seguenti proprietà:

1. $\mathtt{x} + \mathtt{y} \in V$ (chiusura rispetto alla somma)
2. $\alpha\mathtt{x} \in V$ (chiusura rispetto al prodotto)
3. $\mathtt{y} + \mathtt{x} = \mathtt{x} + \mathtt{y}$ (proprietà commutativa)
4. $(\mathtt{x} + \mathtt{y}) + \mathtt{z} = \mathtt{x} + (\mathtt{y} + \mathtt{z})$ (proprietà associativa)
5. $\exists 0 \in V : \mathtt{x} + 0 = \mathtt{x}$ (elemento neutro)
6. $\exists - \mathtt{x} \in V : \mathtt{x} + (\mathtt{x}) = 0$ (inverso additivo)

7. $\alpha(\beta \mathtt{x}) = (\alpha\beta)\mathtt{x}$ (proprietà associativa)

8. $\alpha(\mathtt{x} + \mathtt{y}) = \alpha\mathtt{x} + \alpha\mathtt{y}$ (proprietà distributiva in V)

9. $(\alpha + \beta)\mathtt{x} = \alpha\mathtt{x} + \beta\mathtt{x}$ (proprietà distributiva in $\mathbb{R}$)

10. $1\mathtt{x} = \mathtt{x}$ (elemento neutro)

Si chiama **combinazione lineare** dei vettori $\{\mathtt{x}_1, \dots, \mathtt{x}_k\}$ con coefficienti $\alpha_i \in \mathbb{R}$, $i = 1, \dots, k$ il vettore $\mathtt{x} \in V$ dato da

$$\mathtt{x} = \sum_{i=1}^{k} \alpha_i \mathtt{x}_i$$

sottospazio generato dai vettori $\{\mathtt{x}_1, \dots, \mathtt{x}_k\}$ è formato da tutte le loro combinazioni lineari

$$span(\mathtt{x}_1, \dots, \mathtt{x}_k) := \left\{ \mathtt{x} \in V : \mathtt{x} = \sum_{i=1}^{k} \alpha_i \mathtt{x}_i \right\}.$$

I vettori $\{\mathtt{x}_1, \dots, \mathtt{x}_k\}$ sono **linearmente indipendenti** se

$$\sum_{i=1}^{k} \alpha_i \mathtt{x}_i = 0 \Rightarrow \ \alpha_i = 0, i = 1, \dots, k.$$

Questa condizione equivale al fatto che nessuno dei vettori sia combinazione lineare degli altri. Una `base` è un insieme di vettori linearmente indipendenti tali che ogni vettore dello spazio possa essere espresso come combinazione lineare. La `dimensione` di uno spazio lineare è la cardinalità di una base.

Definizione 2.1.2: Spazio normato

Uno `spazio normato` è uno spazio lineare su cui è definita una funzione

$$\begin{aligned} \|\cdot\| : V &\longrightarrow \mathbb{R} \\ \mathtt{x} &\longmapsto \|\mathtt{x}\| \end{aligned}$$

detta norma, tale che per ogni $\mathtt{x}, \mathtt{y} \in V$ e $\alpha \in \mathbb{R}$ si abbia

1. $\|\mathtt{x}\| \geq 0$ e $\|\mathtt{x}\| = 0 \iff \mathtt{x} = 0$ (positività)

2. $\|\alpha\mathtt{x}\| = |\alpha|\|\mathtt{x}\|$ (omogeneità)

3. $\|\mathtt{x} + \mathtt{y}\| \leq \|\mathtt{x}\| + \|\mathtt{y}\|$ (diseguaglianza triangolare)

uno spazio normato è anche uno **spazio metrico**, in cui distanza tra due vettori è misurata mediante la funzione:

$$d(\mathtt{x}, \mathtt{y}) := \|\mathtt{x} - \mathtt{y}\|.$$

Le più importanti norme in uno spazio $\mathbb{R}^n$ sono:

$$\text{norma due:} \|\mathtt{x}\|_2 = \Big(\sum_{i=1}^{n} \mathtt{x}_i^2\Big)^{\frac{1}{2}}$$

$$\text{norma uno:} \|\mathtt{x}\|_1 = \sum_{i=1}^{n} |\mathtt{x}_i|$$

$$\text{norma } \infty : \|\mathtt{x}\|_\infty = \max_{i=1,\dots,n} |\mathtt{x}_i|$$

Gli spazi funzionali $C[a,b]$ e$L^e[a,b]$ vengono dotati delle norme

$$\|f\|_\infty = \max_{x\in[a,b]} |f(x)|, \quad f \in C[a,b]$$

$$\|g\|_2 = \Big(\int_a^b |g(x)|^2 dx\Big)^{\frac{1}{2}}, \quad g \in L^2[a,b].$$

Un risultato importante che ci servirà di seguito, si veda 6.5 a pagina 40 è il seguente teorema:

Teorema 2.1.1: Equivalenza delle norme

Tutte le norme in $\mathbb{R}^n$ sono equivalenti, nel senso che per ogni coppia di norme $\|\cdot\|_\alpha, \|\cdot\|_\beta$ esistono due costanti positive m e M tali che, per ogni $\mathtt{x} \in \mathbb{R}^n$, si ha:

$$m\|\mathtt{x}\|_\beta \le \|\mathtt{x}\|_\alpha \le M\|\mathtt{x}\|_\beta.$$

Questo teorema consente di studiare indifferentemente qualsiasi norma per la convergenza di una successione di vettori.

Definizione 2.1.3: Spazio di Hilbert

Uno spazio di Hilbert è uno spazio lineare su cui è definito un prodotto scalare (o prodotto interno)

$$\langle\cdot,\cdot\rangle : V \times V \longrightarrow \mathbb{R}$$
$$(\mathtt{x},\mathtt{y}) \longmapsto \langle\cdot,\cdot\rangle$$

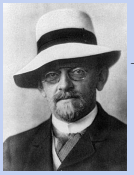
tale che per ogni $\mathtt{x},\mathtt{y},\mathtt{z} \in V$ e $\alpha \in \mathbb{R}$

1. $\langle \mathtt{x},\mathtt{x}\rangle \ge 0$ e $\langle \mathtt{x},\mathtt{x}\rangle = 0 \iff \mathtt{x} = 0$ (positività)
2. $\langle \mathtt{x},\mathtt{y}\rangle = \langle \mathtt{y},\mathtt{x}\rangle$ (proprietà commutativa)
3. $\langle \alpha\mathtt{x},\mathtt{x}\rangle = \alpha\langle \mathtt{x},\mathtt{y}\rangle$ (omogeneità)
4. $\langle \mathtt{x}+\mathtt{y},\mathtt{z}\rangle = \langle \mathtt{x},\mathtt{y}\rangle + \langle \mathtt{y},\mathtt{z}\rangle$ (linearità)

Tale spazio deve essere completo rispetto alla `norma indotta` dal prodotto interno

$$\|\mathtt{x}\| := \langle \mathtt{x},\mathtt{x}\rangle^{\frac{1}{2}}$$

Nota storica 2.1.1: Hilbert



David Hilbert (Königsberg, 23 gennaio 1862 – Gottinga, 14 febbraio 1943) è stato un matematico tedesco. È stato uno dei più eminenti ed influenti matematici del periodo a cavallo tra il XIX secolo e il XX secolo. [Wiki17f]

Due vettori sono `ortogonali` quando il loro prodotto scalare è nullo. Il prodotto scalare comunemente utilizzato in $\mathbb{R}^n$ è: $\langle \mathtt{x},\mathtt{y}\rangle = \sum_{i=1}^n x_i y_i$.

2.2 Matrici

Una matrice $m \times n$ è un quadro di mn numeri reali o complessi di m righe e n colonne. Sia $\mathtt{A} \in \mathbb{C}^{m\times n}$ la matrice **aggiunta** $\mathtt{A}^*$ si ottiene scambiando righe si $\mathtt{A}$ con le sue colonne e coniugandone gli elementi, se $\mathtt{A} \in \mathbb{R}$ allora si dice **trasposta**. Date due matrici $\mathtt{A} \in \mathbb{R}^{m\times n}$ e $\mathtt{B} \in \mathbb{R}^{n\times p}$ il prodotto tra matrici è la matrice $\mathtt{C} = \mathtt{AB} \in \mathbb{R}^{m\times p}$ definita da:

$$c_{ij} = \sum_{k=1}^{n} a_{ik} b_{kj}, \quad i = 1, \dots, m; j = \dots, p.$$

Una matrice reale $m \times n$ rappresenta la generica **trasformazione lineare** tra gli spazi lineari $\mathbb{R}^n$ e $\mathbb{R}^m$. Il Prodotto matriciale corrisponde alla composizione di due trasformazioni lineari.

Una matrice **quadrata** A si dice **invertibile** o **non singolare** se $\exists$ una matrice $\mathtt{A}^{-1}$ detta matrice inversa tale che $\mathtt{A}^{-1}\mathtt{A} = \mathtt{A}\mathtt{A}^{-1} = \mathtt{I}$. Definiamo alcune proprietà delle matrici:

- $(\mathtt{AB})^T = \mathtt{B}^T\mathtt{A}^T$;
- $(\mathtt{AB})^{-1} = \mathtt{B}^{-1}\mathtt{A}^{-1}$;
- $(\mathtt{A}^T)^{-1} = (\mathtt{A}^{-1})^T$;
- $\mathtt{A}$ è invertibile se e solo se ha righe (e colonne) linearmente indipendenti.

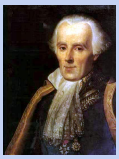
Il **determinante** è una funzione che associa a ciascuna matrice quadrata un numero reale, esso può essere calcolato mediante la formula di Laplace;

$$det(\mathtt{A}) = \sum_{j=1}^{n} (-1)^{i+j} a_{ij} det(A_{ij})$$

essendo A_{ij} la sotto-matrice che si ottiene sa A eliminando la i–esima riga e j–sima colonna. alcune proprietà dei determinanti:

- $det(\mathtt{A}^T) = det(\mathtt{A}), det(\mathtt{A}^*) = \overline{det(\mathtt{A})}, det(\mathtt{A}^{-1}) = det(\mathtt{A})^{-1}$;
- $det(\mathtt{AB}) = det(\mathtt{A})det(\mathtt{B}), det(\alpha\mathtt{A}) = \alpha^n det(\mathtt{A})$;
- uno scambio di due righe produce un cambio di segno del determinante;
- A è non singolare se e solo se $det(\mathtt{A}) \neq 0$

Nota storica 2.2.1: Laplace



Pierre-Simon Laplace, marchese di Laplace (Beaumont-en-Auge, 23 marzo 1749 – Parigi, 5 marzo 1827), è stato un matematico, fisico, astronomo e nobile francese. Fu uno dei principali scienziati nel periodo napoleonico. [Wiki17i]

Definizione 2.2.1: Rango

Il **rango** di una matrice $rank(\mathtt{A})$ può essere definito come il massimo numero di righe (o colonne) linearmente indipendenti o come l'ordine della più grande sotto-matrice quadrata con determinante non nullo (non singolare).

2.2.1 Autovalori e autovettori

Definizione 2.2.2: Autovalore e autovettore

Si dicono `autovalore` ed `autovettore` di una matrice $\mathtt{A}$ uno scalare λ ed un vettore $\mathtt{x} \neq \mathtt{0}$ che verificano la relazione

$$\mathtt{Ax} = \lambda\mathtt{x} \tag{2.1}$$

La 2.1 può essere riscritta:

$$(\mathtt{A} - \lambda\mathtt{I})\mathtt{x} = \mathtt{0}.$$

Questo è un sistema lineare omogeneo dove il vettore $\mathtt{x} \neq \mathtt{0}$, pertanto per ammettere una soluzione non nulla deve annullarsi il determinante della matrice $(\mathtt{A}-\lambda\mathtt{I})$, questo risulta essere un polinomio di grado n in λ, dove n è il numero di righe della matrice $\mathtt{A}$ quadrata. Tale polinomio è noto come `polinomio caratteristico`:

$$p_{\mathtt{A}}(\lambda) = det(\mathtt{A} - \lambda\mathtt{I})$$

quindi calcolare gli zeri di $p_{\mathtt{A}}(\lambda)$ vuol dire calcolare gli n autovalori si $\mathtt{A}$.

La matrice $(\mathtt{A} - \lambda\mathtt{I})$ ha `sempre` $det(\mathtt{A} - \lambda\mathtt{I}) \neq 0$ anche se $\mathtt{A}$ è singolare.

Per ciascun autovalore $\lambda_k, k = 1, \ldots, n$, una soluzione non nulla del sistema singolare omogeneo[1]

$$(\mathtt{A} - \lambda_k\mathtt{I})\mathtt{x} = \mathtt{0}$$

fornisce il corrispondente autovettore, il rango della matrice $(\mathtt{A}-\lambda_k\mathtt{I}) \leq n-1$, dunque l'autovettore rimane determinato a meno di una costante.

Definizione 2.2.3: Spettro

Lo spettro di una matrice è l'insieme degli autovalori

$$\sigma(\mathtt{A}) = \{\lambda_1, \ldots, \lambda_n\} \tag{2.2}$$

Definizione 2.2.4: Raggio spettrale

Il raggio spettrale è il massimo dei moduli degli autovalori

$$\rho(\mathtt{A}) = \max_{k=1,\ldots,n} |\lambda_k|. \tag{2.3}$$

La `molteplicità algebrica` di un autovalore è la sua molteplicità come zero del polinomio caratteristico. Si definisce, `molteplicità geometrica` di un autovalore il massimo numero si autovettori linearmente indipendenti ad esso corrispondenti, in generale per ogni autovalore si ha sempre:

$$\text{molteplicità geometrica} \leq \text{molteplicità algebrica}$$

Se per un autovalore vale il minore stretto, al matrice viene detta `difettiva`.

Proprietà:

[1] Il sistema è singolare $det(\mathtt{A} - \lambda_k\mathtt{I}) = \mathtt{0}$, stiamo cercando soluzioni non banali del vettore $\mathtt{x}$.

- $det(\mathtt{A}) = \prod_{k=1}^{n} \lambda_k$;
- $\sigma(\mathtt{A}^T) = \sigma(\mathtt{A}), \sigma(\mathtt{A}^{-1}) = \{\lambda_1^{-1}, \dots, \lambda_n^{-1}\}, \sigma(\mathtt{A}^p) = \{\lambda_1^p, \dots, \lambda_n^p\}$;
- ad autovalori distinti corrispondono autovettori indipendenti;
- se un autovettore $\mathtt{x}$ è noto il **quoziente di Rayleigh**: $\frac{\mathtt{x}^T\mathtt{A}\mathtt{x}}{\mathtt{x}^T\mathtt{x}}$ fornisce il corrispondente autovalore.

2.3 Matrici di forma particolare

Alcune matrici posseggono una **struttura** quando hanno delle proprietà che rendono più agevole la risoluzione di un problema che le coinvolge, ecco alcune di esse.

Hermitiane

Una matrice è Hermitiana se coincide con la sua aggiunta, se la matrice è reale si dice simmetrica. Una matrice Hermitiana $\mathtt{A}$ è definita positiva se:

$$\mathtt{x}^*\mathtt{A}\mathtt{x} > 0, \quad \forall \mathtt{x} \in \mathbb{C}^n$$

valida per matrici simmetriche reali

Nota storica 2.3.1: Hermite

Charles Hermite (Dieuze, 24 dicembre 1822 – Parigi, 14 gennaio 1901) è stato un matematico francese che diede rilevanti contributi a campi quali teoria dei numeri, forme quadratiche, teoria degli invarianti, polinomi ortogonali, funzioni ellittiche e algebra. Egli fu il primo a dimostrare che la costante e, è un numero trascendente. [Wiki16d].

Teorema 2.3.1: Matrice Hermitiana

Se $\mathtt{A}$ è Hermitiana, i suoi autovalori sono reali ed esiste per $\mathbb{C}^n$ una base si autovettori ortogonali. Se $\mathtt{A}$ è anche definita positiva, gli autovalori sono positivi.

Unitarie

Nota storica 2.3.2: Rayleigh

John William Strutt 3° barone di Rayleigh (Langford Grove, 12 novembre 1842 – Witham, 30 giugno 1919) è stato un fisico britannico. [Wiki16e].

Una matrice complessa è unitaria se la sua inversa coincide con l'aggiunta:

$$\mathtt{Q}\mathtt{Q}^* = \mathtt{Q}^*\mathtt{Q} = \mathtt{I}$$

Una matrice reale è `ortogonale` se l'inverso coincide con la trasposta:

$$\mathtt{Q}^T\mathtt{Q} = \mathtt{Q}\mathtt{Q}^T = \mathtt{I}$$

Teorema 2.3.2

Se $\mathtt{Q}$ è unitaria, allora:

1. $|det(\mathtt{Q})| = 1$, se $\mathtt{Q} \in \mathbb{R} \Rightarrow det(\mathtt{Q}) = \pm 1$;
2. $\|\mathtt{Q}\mathtt{x}\|_2 = \|\mathtt{x}\|_2$, per ogni $\mathtt{x} \in \mathbb{C}^n$.

Triangolari

Una matrice $\mathtt{U}$ si dice triangolare superiore se i suoi elementi verificano $u_{ij} = 0$, per $i > j$. Una matrice $\mathtt{L}$ si dice triangolare inferiore se i suoi elementi verificano $l_{ij} = 0$, per $i < j$. Una matrice $\mathtt{D}$ si dice diagonale se è simultaneamente triangolare superiore e inferiore.

Teorema 2.3.3

Se $\mathtt{T}$ è triangolare il suo determinante è dato dal prodotto degli elementi diagonali. Gli autovalori coincidono con gli elementi diagonali.

Le matrici triangolari formano `un'algebra`: l'inversa di una matrice triangolare, quando esiste, e il prodotto di due matrici triangolari sono ancora matrici triangolari dello stesso tipo. La stessa proprietà vale per matrici unitarie, ma questo caso non si può parlare di algebra dato che l'unitarietà non si trasmette attraverso la somma di matrici ed il prodotto per uno scalare.

Banda

Gli elementi di una matrice a banda verificano la relazione:

$$a_{ij} = 0, \text{ se } i - j \geq k \text{ oppure } i - j \leq -m$$

Sparse

Le matrici sparse sono dotate di pochi elementi diversi da zero, meno del 10%. Pertanto si possono memorizzare solo gli elementi non nulli e della loro posizione all'interno della matrice.

Norme matriciali

Lo spazio lineare $\mathscr{M}_{m\times n}$ cui appartengono le matrici, può essere dotato della struttura di spazio normato definendo uan qualsisi norm $\|\cdot\|$ che verifichi gli assiomi richiesti.

Definizione 2.3.1: Norma indotta

Una norma matriciale si dice **indotta** da una norma vettoriale $\|\cdot\|$, se:

$$\|\mathtt{A}\| = \sup_{\mathtt{x}\neq 0} \frac{\|\mathtt{Ax}\|}{\|\mathtt{x}\|}$$

o se, equivalentemente,

$$\|\mathtt{A}\| = \max_{\|\mathtt{x}\|=1} \|\mathtt{Ax}\|.$$

Una norma così definita viene anche detta **naturale**, o subordinata.

Seguono alcuni teoremi sulle norme matriciali

Teorema 2.3.4: Norma matriciale indotta dalla norma vettoriale ∞

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

Teorema 2.3.5: Norma matriciale indotta dalla norma 1

$$\|\mathtt{A}\|_1 = \max_{i=1,\dots,n} \sum_{i=1}^{m} |a_{ij}|.$$

Teorema 2.3.6: Norma matriciale indotta dalla norma 2

$$\|\mathtt{A}\|_2 = \sqrt{\rho(\mathtt{A}^*\mathtt{A})}.$$

Né consegue che se $\mathtt{A}$ è Hermitiana:

$$\|\mathtt{A}\|_2 = \rho(\mathtt{A}).$$

Numero di condizionamento

Un parametro di fondamentale importanza, per la soluzione di un sistema lineare è il `numero di condizionamento`.

Definizione 2.3.2: Numero di condizionamento

Il numero di condizionamento di una matrice relativamente alla risoluzione di un sistema lineare, la quantità:

$$k(\mathtt{A}) = \|\mathtt{A}\|\|\mathtt{A}^{-1}\|. \tag{2.4}$$

Misura il massimo fattore di amplificazione dell'errore relativo sulla soluzione rispetto all'errore relativo sui dati. Esso è influenzato dalla norma matriciale adottata.

Verifica le seguenti proprietà:

1. per ogni matrice $\mathtt{A}, k(\mathtt{A}) \geq 1$;
2. per ogni norma naturale, $k(\mathtt{I}) = 1$;
3. $k(\mathtt{AB}) \leq k(\mathtt{A})k(\mathtt{B})$;
4. se $\mathtt{Q}$ è ortogonale $k_2(\mathtt{Q}) = 1$;
5. $k_2(\mathtt{A}) = \sqrt{\frac{\lambda_{max}(\mathtt{A}^T\mathtt{A})}{\lambda_{min}(\mathtt{A}^T\mathtt{A})}}$;
6. se A è simmetrica $k_1(\mathtt{A}) = \frac{|\lambda_{max}(\mathtt{A})|}{|\lambda_{min}(\mathtt{A})|}$.

Capitolo 3

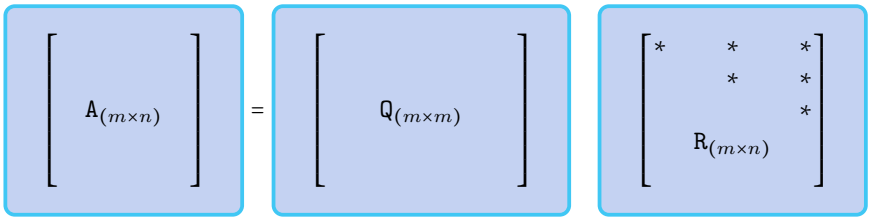
Fattorizzazione QR (anni '60)

Prima di introdurre gli algoritmi dei *metodi iterativi* occorre studiare la fattorizzazione QR.

Teorema 3.0.1: Fattorizzazione QR

Data una matrice **qualsiasi**: $\mathtt{A}$ $m \times n$, $\exists \mathtt{Q} \quad : \mathtt{Q}^T\mathtt{Q} = \mathtt{Q}\mathtt{Q}^T = \mathtt{I}$ (Ortogonale), e $\exists \mathtt{R}$ triangolare superiore $: \mathtt{A} = \mathtt{QR}$

La complessità computazionale di questa fattorizzazione $\mathtt{O}\ (\frac{2}{3}n^3)$[1].



La matrice $\mathtt{R}$ essendo triangolare superiore può essere espressa come: $\mathtt{R}_{(m\times n)} = \begin{bmatrix} \mathtt{R}_{1\,(n\times n)} \\ \mathtt{O}_{(m-n)} \end{bmatrix}$.

Le matrici ortogonali sono *comode* quando si ha a che fare con la norma 2, infatti:

$$\|\mathtt{Qx}\|^2 = (\mathtt{Qx})^T(\mathtt{Qx}) = \mathtt{x}^T\mathtt{Q}^T\mathtt{Qx} = \mathtt{x}^T\mathtt{x} = \|\mathtt{x}\|^2. \tag{3.1}$$

Quando $\mathtt{A}$ è una matrice quadrata non singolare, la fattorizzazione QR può essere utilizzata per la risoluzione del sistema lineare $\mathtt{Ax} = \mathtt{b}$.

Sostituendo ad $\mathtt{A}$ la **sua** QR si ottiene:

$$\begin{cases} \mathtt{Qc} = \mathtt{b} \\ \mathtt{Rx} = \mathtt{c}. \end{cases} \tag{3.2}$$

Grazie alla 3.1 questa fattorizzazione ha un notevole vanataggio sulla crescita del condizionamento.

[1] Peggio di Gauss.

Teorema 3.0.2: Condizionamento QR

Se $\mathtt{A} = \mathtt{QR}$, oppure $\mathtt{A} = \mathtt{RQ}$ allora $\|\mathtt{A}\|_2 = \|\mathtt{R}\|_2$

Dimostrazione. Valendo la 3.1 da $\mathtt{A} = \mathtt{QR}$ otteniamo:

$$\|\mathtt{A}\|_2 \leq \|\mathtt{Q}\|_2 \|\mathtt{R}\|_2 = \|\mathtt{R}\|_2.$$

In maniera analoga partendo da $\mathtt{R} = \mathtt{Q}^T\mathtt{A}$ e dunque:

$$\|\mathtt{R}\|_2 \leq \|\mathtt{Q}\|_2 \|\mathtt{A}\|_2 = \|\mathtt{A}\|_2.$$

Le due relazioni implicano la tesi. □

Teorema 3.0.3: Numero di condizionamento

La matrice $\mathtt{R}$ ha lo stesso numero di condiziomanento rispetto alla norma−2 della matrice $\mathtt{A}$.

Dimostrazione. Poiché $\mathtt{A}^{-1} = \mathtt{R}^{-1}\mathtt{Q}^T$, applicando il teorema 3.0.2 si ottiene:

$$\kappa_2(\mathtt{A}) = \|\mathtt{A}\|_2 \|\mathtt{A}^{-1}\|_2 = \|\mathtt{R}\| \|\mathtt{R}^{-1}\|_2 = \kappa_2(\mathtt{R}).$$

□

Si sono analizzate diverse fattorizzazioni QR: Householder, Givens e Gram-Schmidt. Sono entrambe fondamentali e ampiamente utilizzate nelle simulazioni sperimentali svolte, ma per non dilungarmi eccessivamante sulla trattazione ci si focalizzerà, per ora, solo su Gram-Schmidt. Perché son la **base** degli algoritmi iterativi, che sono stati usati anche in *altri lavori.* **ortogonalizzazione**. Si è **sempre** fatto riferimento al libro di testo [Rod08].

3.1 Ortogonalizzazione di Gram-Schmidt

Il processo di Gram-Schmidt consente di ortogonalizzare n vattori di $\mathbb{R}^m$ con $(m \geq n)$. Inizimo pensando di poter scrivere la matrice $\mathtt{A}$ come composta da vettori colonna:

$$\mathtt{A} = [\mathtt{a}_1, \mathtt{a}_2, \ldots, \mathtt{a}_n]$$

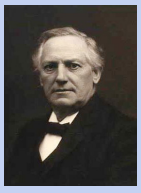
allora questo procedimanto permette di scrivere:

$$\mathtt{Q} = [\mathtt{q}_1, \mathtt{q}_2, \ldots, \mathtt{q}_n]$$

tale che

$$span\{\mathtt{q}_1, \ldots, \mathtt{q}_n\} = span\{\mathtt{a}_1, \ldots, \mathtt{a}_n\} \quad \text{e} \quad \mathtt{q}_i^T\mathtt{q}_j = \delta_{ij}.$$

Nota storica 3.1.1: Gram-Schmidt



Jørgen Pedersen Gram (Nustrup, 27 giugno 1850 – Copenaghen, 29 aprile 1916) è stato un matematico ed attuario danese. Tra i suoi studi si ricordano le espansioni in serie determinate dai metodi dei minimi quadrati, i numeri primi minori di un dato numero e le serie per la funzione zeta di Riemann. Il processo di ortogonalizzazione che porta il suo nome assieme a quello di Erhard Schmidt venne pubblicato per la prima volta nel 1883. Morì a 66 anni dopo essere stato investito da una bicicletta. [Wiki17e]

Erhard Schmidt (Dorpat, 13 gennaio 1876 – Berlino, 6 dicembre 1959) è stato un matematico tedesco. Conseguì il dottorato presso l'Università Georg-August di Gottinga nel 1905 sotto la supervisione di David Hilbert, con cui fornì importanti contributi per l'analisi funzionale. [Wiki17k]

Al primo passo si pone:

$$\tilde{\mathtt{q}}_1 = \mathtt{a}_1, \quad r_{11} = \|\tilde{\mathtt{q}}_1\|_2 \quad \mathtt{q}_1 = \frac{\tilde{\mathtt{q}}_1}{r_{11}}$$

Al generico passo k si calcola

$$\tilde{\mathtt{q}}_k = \mathtt{a}_k - \sum_{j=1}^{k-1} r_{jk}\mathtt{q}_j \tag{3.3}$$

con

$$r_{jk} = \mathtt{q}_j^T \mathtt{a}_k \quad j = 1, 2, \ldots, k-1,$$

ottenendo

$$\mathtt{q}_k = \frac{\tilde{\mathtt{q}}_k}{r_{kk}}, \quad \text{per} \quad r_{kk} = \|\tilde{\mathtt{q}}_k\|.$$

La scelta effettuata per le costanti r_{jk} permette che ogni vettore $\mathtt{q}_k$ risulti ortogonale ai precedenti.

Applicando l'algortimo, pedisseqaumente, per esempio al passo $k = 3$ si ha.

$$k = 3 \quad \tilde{\mathtt{q}}_3 = \mathtt{a}_3 - r_{13}\mathtt{q}_1 - r_{12}\mathtt{q}_2 \tag{3.4}$$

$$r_{13} = \mathtt{q}_1^T \mathtt{a}_3$$

$$r_{12} = \mathtt{q}_1^T \mathtt{a}_2$$

$$\mathtt{q}_3 = \frac{\tilde{\mathtt{q}}_3}{r_{33}} \tag{3.5}$$

e unendo le 3.4 e 3.5 si ottiene:

$$\mathtt{q}_3 = \frac{\mathtt{a}_3 - r_{13}\mathtt{q}_1 - r_{12}\mathtt{q}_2}{r_{33}} \Rightarrow \mathtt{a}_3 = r_{13}\mathtt{q}_1 + r_{12}\mathtt{q}_2 + r_{13}\mathtt{q}_3,$$

che possiamo riscrivere come:

$$\mathtt{a}_k = \sum_{j=1}^{k} r_{jk}\mathtt{q}_j, \quad k = 1, \ldots, n \tag{3.6}$$

e dunque la matrice `A` può essere rappresentata come:

$$\mathtt{A} = [\mathtt{q}_1 \ldots \mathtt{q}_k \ldots \mathtt{q}_n] \begin{bmatrix} r_{11} & \ldots & r_{ik} & \ldots & r_{1n} \\ & \ddots & \vdots & & \vdots \\ & & r_{kk} & & \vdots \\ & & & \ddots & \vdots \\ & & & & r_{nn} \end{bmatrix}.$$

L'ortogonalizzazione delle colonne della matrice `A` costruisce la fattorizzazione QR. Questo procedimento è noto come **CGS** (*Classical Gram-Schmidt*). La matrice `R` è costruita per colonne e quando $n < m$ la matrice `Q` pur avendo colonne ortonormali, **non è** ortogonale, non essendo quadrata. Può essere completata per ottenere una matrice ortogonale aggiungendo $m - n$ vettori che completino la base ortonormale.

È stata sudiata una una differente versione dell'algoritmo, perché la colonne di `Q` tendono a perdere la reciproca ortogonalità al procedere delle iterazioni, rendendo numericamente instabile l'algortimo.

La versione prende il nome di **MGS** (*Modified Gram-Schmidt*), genera una successione di $n+1$ matrici tali che $\mathtt{A}^{(1)} = \mathtt{A}$ e $\mathtt{A}^{(n+1)} = \mathtt{Q} \in \mathbb{R}^{m\times n}$.

La k–esima matrice assume la forma

$$\mathtt{A}^{(k)} = [\mathtt{q}_1, \ldots, \mathtt{q}_{k-1}, \mathtt{a}_k^{(k)}, \ldots, \mathtt{a}_n^{(k)}]$$

con

$$\begin{cases} \mathtt{q}_i^T \mathtt{q}_j = \delta_{ij}, \ i, j = 1, \ldots, k-1, \\ \mathtt{q}_j^T \mathtt{a}_j^{(k)} = 0, \ i = 1, \ldots, k-1; \ j = k, \ldots, n. \end{cases}$$

Al passo k, la matrice $\mathtt{A}^{(k+1)}$ viene ottenuta attraverso la seguente trasformazione:

$$[\mathtt{q}_1, \ldots, \mathtt{q}_{k-1}, \mathtt{a}_k^{(k)}, \ldots, \mathtt{a}_n^{(k)}] \to [\mathtt{q}_1, \ldots, \mathtt{q}_k, \mathtt{a}_{k+1}^{(k+1)}, \ldots, \mathtt{a}_n^{(k+1)}]$$

ponendo

$$\begin{aligned} r_{kk} &= \|\mathtt{a}_k^{(k)}\|, \\ \mathtt{q}_k &= \frac{\mathtt{a}_k^{(k)}}{r_{kk}}, \\ r_{kj} &= \mathtt{q}_k^T \mathtt{a}_j^{(k)}, \quad j = k+1, \ldots, n, \\ \mathtt{a}_j^{(k+1)} &= \mathtt{a}_j^{(k)} - r_{jk}\mathtt{q}_k, \quad j = k+1, \ldots, n. \end{aligned}$$

Si ottengono due famiglie di vettori:

1. $\{\mathtt{q}_1, \ldots, \mathtt{q}_k\}$ formata da vettori tra loro ortonormali,
2. $\{\mathtt{a}_{k+1}^{(k+1)}, \ldots, \mathtt{a}_n^{(k+1)}\}$ gli elementi sono ortogonali a tutti i vettori della prima famiglia.

All'n–esima iterazione la prima famiglia conterrà n vettori tra loro ortonormali, la seconda sarà vuota. L'algoritmo termina n passi, la matrice `R` viene costruita per righe, anzichè per colonne. La complessità computazionale è $O(mn^2)$. Anche in **MGS** può verificarsi una perdita di ortogonalità nelle colonne di `A`.

Capitolo 4

Sistemi sovradeterminati e sottoderminati

4.1 Classificazione dei sistemi

PONIAMOCI il seguente problema:

$$\mathtt{A}\mathtt{x} = \mathtt{b}, \quad \mathtt{A} \in \mathbb{R}^{n \times n}, \quad \mathtt{b}, \mathtt{x} \in \mathbb{R}^n,$$
$$k = rank(\mathtt{A}) = n$$

esso ammette soluzione, se e solo se, il vettore `b` appartiene allo spazio lineare generato dalle colonne di `A`, si veda la figura 4.1.

Partendo da uno spazio vettoriale, il `dominio`, di dimensione n, attraverso l'operatore `A`, cerchiamo le `x` nel `codominio`, spazio vettoriale di dimensione $\mathbb{R}^n$.

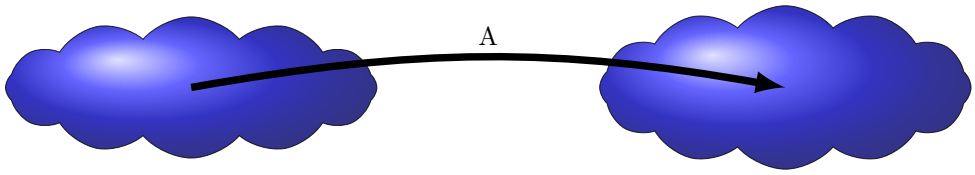


Figura 4.1: Dominio e codominio.

Nel caso in cui il domino o il codominio non hanno più dimensione $\mathbb{R}^n$, la soluzione potrebbe non esistere o esisterne infinite. Inoltre il problema dipende dal rango della matrice `A` che potrebbe non essere pieno ovvero:

$$k = rank(\mathtt{A}) < \min(m, n).$$

$k < n < m$ caso sovraderminato

Il rango della matrice `A` indica quante colonne sono linearmente dipendenti, se riesco a conoscere quali sono, posso pensare di eliminarle, per farlo pongo a 0 le corrispondenti componenti di `x`,

ma questa operazione potrebbe non essere semplice. Per esempio una matrice `A` ha due colonne linearmente dipendenti[1], il vettore `x` risulta allora:

$$\begin{bmatrix} \spadesuit & \spadesuit & \spadesuit & \star & \star \\ \spadesuit & \spadesuit & \spadesuit & \star & \star \\ \spadesuit & \spadesuit & \spadesuit & \star & \star \\ \spadesuit & \spadesuit & \spadesuit & \star & \star \\ \spadesuit & \spadesuit & \spadesuit & \star & \star \\ \spadesuit & \spadesuit & \spadesuit & \star & \star \\ \spadesuit & \spadesuit & \spadesuit & \star & \star \end{bmatrix} \begin{bmatrix} \spadesuit \\ \spadesuit \\ \spadesuit \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} \spadesuit \\ \spadesuit \\ \spadesuit \\ \spadesuit \\ \spadesuit \\ \spadesuit \\ \spadesuit \end{bmatrix}$$

$k < m < n$ caso sottodeterminato

Posso avere ∞ soluzioni.
Esempio: dato il sistema lineare:

$$\begin{cases} 2x + y + z = 4 \\ 4x + 2y + 2z = 8 \end{cases} \Rightarrow \begin{bmatrix} 2 & 1 & 1 \\ 4 & 2 & 2 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} 4 \\ 8 \end{bmatrix}$$

le due equazioni sono linearmente dipendenti, dato che la seconda è il doppio della prima, quindi abbiamo ∞ soluzioni, ma il rango vale 1 dunque sono **ancora più** ∞ soluzioni. Ipotizziamo ora di cambiare il secondo membro della seconda equazione[2]:

$$\underbrace{4x + 2y + 2z}_{\text{l.d.}} = \underbrace{5}_{\text{l.i.}}$$

si divide i due parti: una linearmente dipendente e una linearmente indipendente, si è dunque passati da ∞ soluzioni a `nessuna soluzione`.

$k < m = n$

Il sistema è quadrato, ma non a rango pieno allora si possono verificare i casi:

1. nessuna soluzione, poiché si arriva ad una palese incongruenza:

$$\begin{cases} x + 2y = 1 \\ 2x + 4y = 3 \end{cases} \Rightarrow \begin{bmatrix} 1 & 2 \\ 2 & 4 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} 1 \\ 3 \end{bmatrix};$$

2. ∞ soluzioni

$$\begin{cases} x + 2y = 1 \\ 2x + 4y = 2 \end{cases} \Rightarrow \begin{bmatrix} 1 & 2 \\ 2 & 4 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} 1 \\ 2 \end{bmatrix}.$$

[1] Le ultime due colonne in $\star$.
[2] Ora non sono più una il doppio dell'altra.

Allora possiamo pensare di procedere lavorando con una matrice a rango ridotto: k. Questo tipo di problema è noto come **rank deficient**. Questo porta ad avere k autovalori: $\lambda_1, \lambda_2, \ldots, \lambda_k$, gli altri sono **nulli**[3], ma facendo `calcolo numerico` essendoci errori, potrebbe non essere $0 \Rightarrow$ il concetto di rango dipende dalla precisione:

$$\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_k > \epsilon > \lambda_{k+1} \geq \cdots \geq \lambda_n$$

quindi il problema si sposta su **quanto** posso scegliere questo errore, in alcuni casi potrebbe essere semplice, in altri impossibile.

Definizione 4.1.1: range di una matrice o immagine

Il range rappresenta il codominio della funzione:

$$R(\mathtt{A}) = \left\{ \mathtt{y} \in \mathbb{R}^m : \mathtt{y} = \sum_{j=1}^{n} x_j \mathtt{a}_j \right\} \subset \mathbb{R}^m$$

Definizione 4.1.2: nucleo di una matrice

$$N(\mathtt{A}) = \{ \mathtt{x} \in \mathbb{R}^n : \mathtt{Ax} = \mathtt{0} \} \subset \mathbb{R}^n$$

entrambi sono sottospazi.

4.2 Caso *facile* $m > n = k$ sovradeterminato rango pieno

Dato il sistema lineare

$$\mathtt{A}_{(n\times m)}\mathtt{x}_{(m\times 1)} = \mathtt{b}_{n\times 1} \tag{4.1}$$

Il caso è sovraderminato a rango pieno: $rank(\mathtt{A}) = n$. Ovvero $\nexists$ soluzione. Se non posso rendere *uguali* i vettori `Ax` e `b`, allora cerco di renderli *quasi uguali*, per far ciò utilizzo la norma, la norma è *qualcosa* che mi dice quanto lungo è un vettore[4].

Se i vettori non possono essere uguali, né cerco la minima distanza. Ovvero cerco di risolvere un problema del tipo:

$$\min_{x\in\mathbb{R}^n} \|\mathtt{Ax} - \mathtt{b}\|_2 \tag{4.2}$$

il pedice 2 indica che è la norma euclidea. La norma euclidea è così definita:

$$\|\mathtt{x}\| = \sqrt{\sum_{i=1}^{n} x_i^2} = \sqrt{\mathtt{x}^T\mathtt{x}} \Rightarrow \tag{4.3}$$

$$\|\mathtt{Ax} - \mathtt{b}\| = \sqrt{\sum_{i=1}^{m} [(\mathtt{Ax})_i - \mathtt{b}_i]^2}$$

[3] O comunque molto piccoli.
[4] Infatti è uno scalare.

la radice è sempre scomoda, eliminiamola facendo il quadrato, minimizzeremo appunto la norma al quadrato. Ricordando $(\mathtt{Ax})_i = \sum_{j=1}^{n} a_j x_j$ sostituendo ed eliminando la radice:

$$\|\mathtt{Ax} - \mathtt{b}\|^2 = \sum_{i=1}^{m}[\sum_{j=1}^{n} a_j x_j - \mathtt{b}_i]^2 = \varphi(x_1, x_2, \ldots, x_n)$$

Studiando geometria di questa funzione abbiamo: in 1 variabile è una parabola, in 2 variabili è un paraboloide, `sicuramente` φ ha `un solo` minimo. Da un problema ad ∞ soluzioni passo a un problema a `una ed una sola soluzione` è dunque ben posto, questo tipo di problemi prende il nome: `LSP (Least Square Problem)`. Come lo risolviamo? Per trovare un minimo occorre annullare la derivata, pertanto deriviamo i singoli termini.

Richiami

Ricordando alcune formule che saranno utili di seguito, consideriamo la seguente funzione $f(\mathtt{x}) = \mathtt{b}^T\mathtt{x}$, con $\mathtt{b}$ vettore di costanti, $\mathtt{x}$ vettore di variabili, possiamo certamente scrivere

$$\mathtt{b}^T\mathtt{x} = \sum_{i=1}^{n} b_i x_i = \mathtt{x}^T\mathtt{b}$$

se deriviamo questa espressione otteniamo

$$\begin{aligned} &\frac{\partial}{\partial x_k}\sum_{i=1}^{n} b_i x_i = \mathtt{b}_k \Rightarrow \\ &\nabla\mathtt{b}^T\mathtt{x} = \mathtt{b} = \nabla\mathtt{x}^T\mathtt{b} \end{aligned} \tag{4.4}$$

dove l'operatore ∇ indica le derivate parziali per ogni variabile x_i.

Consideriamo ora questa matrice quadrata: $\mathtt{x}^T\mathtt{Ax}$, ricordando che il prodotto matrice vettore `è non` commutativo, ma associativo possiamo *mettere* le parentesi come meglio riteniamo e sviluppare:

$$\begin{aligned} \mathtt{x}^T(\mathtt{Ax}) &= \sum_{i=1}^{n} x_i(\mathtt{Ax})_i = \\ &= \sum_{i=1}^{n} x_i(\sum_{j=1}^{n} a_{ij}x_j) = \text{tutti scalari} \Rightarrow \\ &= \sum_{i=1}^{n}\sum_{j=1}^{n} a_{ij}x_i x_j. \end{aligned} \tag{4.5}$$

Deriviamo la 4.5

$$\begin{aligned} \frac{\partial}{\partial x_k}\sum_{i=1}^{n}\sum_{j=1}^{n} a_{ij}x_i x_j &= \sum_{j=1}^{n} a_{kj}x_j + \sum_{i=1}^{n} a_{ik}x_i = \\ &= (\mathtt{Ax})_k + (\mathtt{A}^T\mathtt{x})_k \end{aligned} \tag{4.6}$$

i risultati ottenuti di nostro interesse sono la 4.2 4.6 che riportiamo. Che per una matrice simmetrica diventano:

$$\mathtt{A} = \mathtt{A}^T$$
$$\nabla(\mathtt{x}^T\mathtt{A}\mathtt{x}) = \mathtt{A}\mathtt{x} + \mathtt{A}^T\mathtt{x} = 2\mathtt{A}\mathtt{x} \tag{4.7}$$
$$\nabla\mathtt{b}^T\mathtt{x} = \nabla\mathtt{x}^T\mathtt{b} = \mathtt{b}. \tag{4.8}$$

4.2.1 Sistema delle equazioni normali

Ricordando la 4.3 eseguiamo il quadrato della norma:

$$\|\mathtt{A}\mathtt{x} - \mathtt{b}\|^2 = (\mathtt{A}\mathtt{x} - \mathtt{b})^T(\mathtt{A}\mathtt{x} - \mathtt{b}) = [(\mathtt{A}\mathtt{x})^T - \mathtt{b}^T][\mathtt{A}\mathtt{x} - \mathtt{b}] =$$
$$= [\mathtt{x}^T\mathtt{A}^T - \mathtt{b}^T][\mathtt{A}\mathtt{x} - \mathtt{b}] = \mathtt{x}^T \underbrace{\mathtt{A}^T\mathtt{A}}_{simm.}\,\mathtt{x} \underbrace{-\mathtt{x}^T\mathtt{A}^T\mathtt{b} - \mathtt{b}^T\mathtt{A}\mathtt{x}}_{sonouguali} + \underbrace{\mathtt{b}^T\mathtt{b}}_{costante} =$$
$$= \mathtt{x}^T\mathtt{A}^T\mathtt{A}\mathtt{x} - 2\mathtt{x}^T\mathtt{A}^T\mathtt{b} + \mathtt{b}^T\mathtt{b} = \varphi(\mathtt{x}).$$

$\mathtt{x}^T\mathtt{A}^T\mathtt{b}$ e $\mathtt{b}^T\mathtt{A}\mathtt{x}$ sono uguali perché: $(\mathtt{b}^T\mathtt{A}\mathtt{x})^T = \mathtt{x}^T\mathtt{A}^T\mathtt{b}$, essendo scalari, il trasposto di uno scalare è lo scalare stesso. Ora annulliamo e deriviamo $\varphi(\mathtt{x})$, ricordando le regole 4.2 e 4.6:

$$\nabla\varphi(\mathtt{x}) = 0$$
$$\cancel{2}\mathtt{A}^T\mathtt{A}\mathtt{x} - \cancel{2}\mathtt{A}^T\mathtt{x} = 0 \Rightarrow$$
$$\boxed{\mathtt{A}^T\mathtt{A}\mathtt{x} = \mathtt{A}^T\mathtt{b}} \tag{4.9}$$

siamo giunti al `Sistema delle equazioni normali (ortogonali)` la 4.9 che gode delle seguenti:

1. è un sistema lineare
2. quadrato, infatti:
   - $\mathtt{A}, \in \mathbb{R}^{m\times n}$
   - $\mathtt{A}^T, \in \mathbb{R}^{n\times m}$
   - $\mathtt{A}^T\mathtt{A}, \in \mathbb{R}^{n\times n}$
   - $\mathtt{b}, \in \mathbb{R}^m$
   - $\mathtt{A}^T\mathtt{b}, \in \mathbb{R}^n$
3. e la matrice $\mathtt{A}^T\mathtt{A}$ ha $k = rank(\mathtt{A}^T\mathtt{A}) = \min(n, n) = n \Rightarrow$ è non singolare

La matrice $\mathtt{A}^T\mathtt{A}$ è:

1. quadrata $n \times n$
2. simmetrica $\Rightarrow$ gli autovalori $\in \mathbb{R}$.

3. definita positiva, ovvero: $\mathtt{x}^T(\mathtt{A}^T\mathtt{A})\mathtt{x} > 0, \forall \mathtt{x} \neq \mathtt{0}$ questo ci permette di affermare che gli autovalori sono maggiori di 0, e che la matrice può essere invertita.

Possiamo risolvere la 4.9, come un normale sistema, soluzione delle equazioni normali: Least Square Solution (LSS):

$$\mathtt{x} = (\mathtt{A}^T\mathtt{A})^{-1}\mathtt{A}^T\mathtt{b}. \tag{4.10}$$

La matrice $\mathtt{A}^\dagger = (\mathtt{A}^T\mathtt{A})^{-1}\mathtt{A}^T$ viene detta `pseudo-inversa` è la matrice risolvente del problema ai minimi quadrati, è inversa *solo* a sinistra:

$$\mathtt{A}^\dagger\mathtt{A} = (\mathtt{A}^T\mathtt{A})^{-1}(\mathtt{A}^T\mathtt{A}) = \mathtt{I}$$
$$\mathtt{A}\mathtt{A}^\dagger = \mathtt{A}(\mathtt{A}^T\mathtt{A})^{-1}\mathtt{A}^T \neq \mathtt{I}$$

trovare la soluzione coma nella 4.10 non è in generale facile numericamente, si preferisce scomporre la matrice, o meglio fattorizzarla, per cercare di ottenere sistemi più facili.

Fattorizzazione Cholesky

Teorema 4.2.1: Cholesky

Se $\mathtt{B}$ è simmetrica definita positiva $\Rightarrow \exists$ una unica matrice $\mathtt{L}$ triangolare inferiore : $\mathtt{B} = \mathtt{L}\mathtt{L}^T$. Questa è nota come *fattorizzazione di Cholesky*.

Nota storica 4.2.1: Cholesky

André-Louis Cholesky (Montguyon, 15 ottobre 1875 – Bagneux, 31 agosto 1918) è stato un ingegnere francese. Divenne Ufficiale di artiglieria nella prima guerra mondiale, trovò un modo per migliorare i procedimenti per il calcolo di minimi quadrati definendo il procedimento di decomposizione di matrici. Muore per le ferite riportate sul campo di battaglia. Il suo procedimento venne pubblicato solo nel 1924 da un ufficiale di nome Benoit. [Wiki16b]. Il Prof. Claude Brezinski[a] ha contattato un nipote di Cholesky riuscendo a visionare gli appunti originali, questi risultano uguali a quelli sviluppati successivamente dai matematici.

[a] Collega del Prof. Rodriguez?

Abbiamo visto che nella 4.9 la matrice $\mathtt{A}^T\mathtt{A}$ è definita positiva, pertanto è possibile applicare il teorema 4.2.1, questo ci permette di scrivere: $\mathtt{B} = \mathtt{A}^T\mathtt{A} := \mathtt{L}\mathtt{L}^T \Rightarrow$

$$\underbrace{\mathtt{L}(\mathtt{L}^T\mathtt{x})}_{\mathtt{y}} = \underbrace{\mathtt{A}^T\mathtt{b}}_{\mathtt{c}} \tag{4.11}$$

l'equazione 4.11 può essere scomposta in un sistema:

$$\begin{cases} \mathtt{L}\mathtt{y} = \mathtt{c} \\ \mathtt{L}^T\mathtt{x} = \mathtt{y} \end{cases} \tag{4.12}$$

Analizzando il prodotto matriciale otteniamo:

$$\mathtt{A} = \mathtt{L}\mathtt{L}^T = \begin{bmatrix} l_{11} & & & & \\ \vdots & \ddots & & & \\ l_{1i} & \dots & l_{ii} & & \\ \vdots & & & \ddots & \\ l_{1n} & \dots & \dots & \dots & l_{nn} \end{bmatrix} \begin{bmatrix} l_{1n} & \dots & l_{1j} & \dots & l_{1n} \\ & \ddots & \vdots & & \vdots \\ & & l_{ll} & & \vdots \\ & & & \ddots & \vdots \\ & & & & l_{nn} \end{bmatrix}$$

eseguendo questo prodotto si arriva:

$$a_{ij} = \sum_{k=1}^{i} l_{ki} l_{kj}, \quad i \leq j$$
$$a_{ij} = l_{jj}^2 + \sum_{k=1}^{i-1} l_{ki} l_{kj}, \quad i \leq j.$$

avendo esplicitato l'ultimo termine: $i = j$ gli elementi di $\mathtt{L}$ in funzione di $\mathtt{A}$, e *separando* i due casi:$i < j$, $i = j$, pertanto:

$$l_{ij} = \frac{1}{r_{ii}} \Big(a_{ij} - \sum_{k=1}^{i-1} l_{ki} l_{kj} \Big), \quad i < j,$$
$$l_{jj} = \Big(a_{jj} - \sum_{k=1}^{j-1} r_{kj}^2 \Big)^{\frac{1}{2}}, \quad i = j.$$

Il fatto che la matrice sia simemtrica definita positiva garantisce che il radicando risulti positivo. Il costo computazionale della fattorizzazione di Cholesky è pari a $\mathtt{O}\ (\frac{n^3}{6})$[5]. Lo svantaggio di questo approccio riguarda la **stabilità**, infatti la matrice $\mathtt{A}^T\mathtt{A}$ ha un numero di condizionamento pari al quadrato di $\mathtt{A} \Rightarrow k(\mathtt{A}^T\mathtt{A}) = (k(\mathtt{A}))^2$, e ricordando che:

$$\frac{\|\delta \mathtt{x}\|}{\|\mathtt{x}\|} \leq \kappa(\mathtt{A}) \frac{\|\delta \mathtt{b}\|}{\|\mathtt{b}\|}$$
$$\kappa(\mathtt{A}) = \|\mathtt{A}\| \|\mathtt{A}\|^{-1} \geq 1$$

dato che il numero di condizionamento di una matrice è sempre $\kappa(\mathtt{A}) \geq 1$ il quadrato sarà ancora peggiore, questo vuol dire che *l'operatore* amplifica maggiorante gli errori, potrebbe portare a dei valori completamente inutilizzabili, cosa che non accade con la fattorizzazione QR, come visto precedentemente.

Quindi riassumendo:

Algoritmo: Cholesky
Input: $A^T A$
Output: minima soluzione

1. $\mathtt{A}^T\mathtt{A} = \mathtt{L}\mathtt{L}^T$
2. risolvi l'equazione $\mathtt{L}(\mathtt{L}^T\mathtt{x}) = \mathtt{A}^T\mathtt{b}$
3. ovvero il sistema:
$$\begin{cases} \mathtt{L}\mathtt{y} = \mathtt{c} \\ \mathtt{L}^T\mathtt{x} = \mathtt{y} \end{cases}$$

[5]Quello di Gauss è dell'ordine $\mathtt{O}\ (\frac{n^3}{3})$, il doppio di Cholesky.

Si può migliorare . . .

Ricordando la 4.2, vogliamo *minimizzare* il quadrato della norma 2 di una matrice $m \times n$, cosa che possiamo fare applicando il teorema 3.0.1 a pagina 13, che si utilizza per questo tipo di matrici, e dunque fattorizzando QR. Si ha pertanto: $A = QR$:

$$\|Ax - b\|^2 = \|QRx - b\|^2 =$$
$$= \|Q(Rx - Q^T b)\|^2 = \|Rx - Q^T b\|^2 =$$
$$= \left\| \begin{bmatrix} R_1 \\ 0 \end{bmatrix} x - \begin{bmatrix} Q_1^T \\ Q_2^T \end{bmatrix} b \right\|^2 = \left\| \begin{bmatrix} R_1 x \\ 0 \end{bmatrix} - \begin{bmatrix} Q_1^T b \\ Q_2^T b \end{bmatrix} \right\|^2 =$$
$$= \|R_1 x - \underbrace{Q_1^T b}_{c_1}\|^2 + \|\underbrace{Q_2^T b}_{c_2}\|^2.$$

minimizzando:

$$\min_x \|Ax - b\|^2 = \min_x \|R_1 x - c_1\|^2 + \|c_2\|^2 \tag{4.13}$$

minimizzare vuol dire trovare la x tale che $R_1 x - c_1 = 0 \Rightarrow R_1 x = c_1$, allora la 4.13 diventa: $\min_x \|Ax - b\|^2 = \|c_2\|^2$ che risulta il minimo. Riassumendo:

Algoritmo: Fattorizzazione QR
Input: A
Output: minima soluzione

1. $A = QR$
2. $R = \begin{bmatrix} R_1 \\ 0 \end{bmatrix}$ $c_1 = Q_1^T b$ $c_2 = Q_2^T b$
3. risolvo $R_1 x = c_1 \rightarrow x$ **teoricamente** coincide con la 4.10, nella pratica no, perché sono presenti gli errori
4. $\min \|Ax - b\|^2 = \|c_2\|^2$.

Il vantaggio di questa fattorizzazione rispetto al metodo di Cholesky e che il condizionamento rimane lineare, non va al quadrato.

4.3 $k = m < n$

Teorema 4.3.1

$S = \{ x \in \mathbb{R}^n : \| Ax - b\|^2 = \min \}$ $x \in S \iff A^T(\underbrace{b - Ax}_{r=residuo}) = 0$ $A^T b - A^T A x = 0 \Rightarrow A^T A x = A^T b$

Il teorema 4.3.1 ci assicura che il minimo può essere trovato come $\mathtt{A}^T\mathtt{Ax} = \mathtt{A}^T\mathtt{b}$[6].

Data la matrice $\mathtt{A} \in \mathbb{R}^{n\times m}$, il suo rango vale $m = \min(n,m)$, il prodotto $\mathtt{A}^T\mathtt{A}$ ha dimensione $n \times n$, dunque ha rango: $k = rank(\mathtt{A}^T\mathtt{A}) = \min(n,n) = n$[7]. Quindi certamente $\mathtt{A}^T\mathtt{A}$ è singolare. Il sistema è consistente dunque posso `sempre` raggiungere l'uguaglianza, ovvero ammette sempre una soluzione. La soluzione $\mathtt{r}$ dell'equazione $\mathtt{A}^T(\mathtt{b}-\mathtt{Ax}) = \mathtt{A}^T\mathtt{r} = \mathtt{0}$ equivale al nucleo di $\mathtt{A}^T$, infatti:

$$\mathscr{N}(\mathtt{A}^T) = \left\{ \mathtt{x} \in \mathbb{R}^m : \mathtt{A}^T\mathtt{x} = \mathtt{0} \right\} \Rightarrow \mathtt{r} \in \mathscr{N}(\mathtt{A}^T).$$

il vettore residuo è per definizione pari:

$$\mathtt{r} = \mathtt{b} - \mathtt{Ax} \Rightarrow \mathtt{b} = \underbrace{\mathtt{Ax}}_{\in \mathscr{R}(\mathtt{A})} + \underbrace{\mathtt{r}}_{\in \mathscr{N}(\mathtt{A}^T)} \tag{4.14}$$

i due vettori sono perpendicolari:

$$\mathtt{Ax} \perp \mathtt{r}$$

il codominio, vettore $\mathtt{b}$, $\in \mathbb{R}^m$, mentre $\mathtt{Ax} \in \mathscr{R}(\mathtt{A})$, quindi:

$$\mathbb{R}^m = \mathscr{R}(\mathtt{A}) \oplus \mathscr{N}(\mathtt{A}^T)$$

dove $\oplus$ indica la somma diretta:

$$\oplus \equiv \begin{cases} \mathscr{R}(\mathtt{A}) \bigcup \mathscr{N}(\mathtt{A}^T) = \mathbb{R}^m \\ \mathscr{R}(\mathtt{A}) \perp \mathscr{N}(\mathtt{A}^T). \end{cases}$$

Il problema è che abbiamo ∞ vettori che soddisfano la 4.14, allora cerchiamo il più corto. In sostanza cerchiamo soluzioni:

$$\begin{aligned} &\min \|\mathtt{x}\| \\ &s.a.\ \mathtt{A}^T\mathtt{Ax} = \mathtt{A}^T\mathtt{b}. \end{aligned}$$

Il problema è noto come Minimum Norm Least Squares Solution (MNLSS) o LSS duale.

Analizzando i problema da più vicino, sostituendo $\mathtt{y} = \mathtt{Az}$ può essere riscritto:

$$\begin{cases} \min \|\mathtt{y}\|_2^2 \\ s.a.\ \mathtt{A}^T\mathtt{y} = \mathtt{c} \end{cases} \quad \begin{cases} \min \frac{1}{2}\mathtt{y}^T\mathtt{y} \\ s.a.\ \mathtt{A}^T\mathtt{y} = \mathtt{c} \end{cases}$$

con $\mathtt{y} \perp \mathscr{N}(\mathtt{A})$, inoltre il fattore $\frac{1}{2}$, è è utile numericamente. Per rispettare i vincoli scriviamo e risolvere il problema scriviamo i moltiplicatori di Lagrange, che sono tanti quanti sono le equazioni del sistema, con $z \in \mathbb{R}^n$, calcolo la distanza dall'origine:

$$\begin{aligned} \mathscr{L}(\mathtt{y},\mathtt{z}) &= \frac{1}{2}\mathtt{y}^T\mathtt{y} - \sum z_i[(\mathtt{A}^T\mathtt{y})_i - \mathtt{c}_i] = \\ &= \frac{1}{2}\mathtt{y}^T\mathtt{y} + \mathtt{z}^T(\mathtt{c} - \mathtt{A}^T\mathtt{y}) = \\ &= \frac{1}{2}\mathtt{y}^T\mathtt{y} + \mathtt{z}^T\mathtt{c} - \mathtt{z}^T\mathtt{A}^T\mathtt{y}. \end{aligned}$$

[6] Che coincide con la 4.9.

[7] Si faccia riferimento al punto 3 a pagina 21, il rango della matrice per quel caso è n, per questo caso è sempre n, ma i due ranghi sono diversi, infatti per quel caso 3 a pagina 21 $n < m$, in questo caso $n > m$, quindi `questa` n è maggiore di quel caso. Per essere chiari se chiamassi n_1 il rango della matrice nel caso 3 a pagina 21, e n_2 il rango in studio possiamo scrivere $n_2 > n_1$.

derivando, ricordando le opportune regole di derivazione per matrici 4.8 a pagina 21 e imponendole nulle le derivate:

$$\nabla_y \mathscr{L}(\mathtt{y}, \mathtt{z}) = \frac{1}{\cancel{2}}\cancel{2}\mathtt{y} - \mathtt{A}\mathtt{z} = \mathtt{0}$$
$$\nabla_z \mathscr{L}(\mathtt{y}, \mathtt{z}) = \mathtt{c} - \mathtt{A}^T\mathtt{y} = \mathtt{0}$$

quindi si ottiene:

$$\begin{cases} \mathtt{A}^T\mathtt{A}\mathtt{z} = \mathtt{c} \\ \mathtt{y} = \mathtt{A}\mathtt{z}. \end{cases} \tag{4.15}$$

Invertendo la seconda equazione delle 4.15 si ottiene:

$$\mathtt{z} = \underbrace{\mathtt{A}(\mathtt{A}^T\mathtt{A})^{-1}}_{(\mathtt{A}^T)^\dagger}\mathtt{c}$$

dove $(\mathtt{A}^T)^\dagger = \mathtt{A}(\mathtt{A}^T\mathtt{A})^{-1}$ è la matrice pseudo-inversa duale che è solo inversa destra:

$$\mathtt{A}^T(\mathtt{A}^T)^\dagger = \mathtt{A}^T\mathtt{A}(\mathtt{A}^T\mathtt{A})^{-1} = \mathtt{I}$$
$$(\mathtt{A}^T)^\dagger\mathtt{A}^T = \mathtt{A}(\mathtt{A}^T\mathtt{A})^{-1}\mathtt{A}^T \neq \mathtt{I}.$$

Capitolo 5

Autovalori e autovettori

CONSIDERIAMO il seguente problema[1]

$$\begin{aligned} \mathtt{Ax} = \lambda \mathtt{x} \quad & \mathtt{A} \in \mathbb{R}^{n \times n}, \mathtt{x} \in \mathbb{R}^n \backslash \{0\}, \lambda \in \mathbb{R} \\ & \mathtt{Ax} - \lambda \mathtt{x} = \mathtt{0} \\ & \mathtt{Ax} - \lambda \mathtt{Ix} = \mathtt{0} \\ & (\mathtt{A} - \lambda \mathtt{I})\mathtt{x} = \mathtt{0} \end{aligned}$$

dove λ è detto autovalore della, matrice $\mathtt{A}$; il sistema lineare omogeneo è soddisfatto se la matrice $(\mathtt{A} - \lambda \mathtt{I})$ è singolare ovvero:

$$\begin{aligned} det(\mathtt{A} - \lambda \mathtt{I}) &= \mathtt{0} \\ p_n(\lambda) &= det(\mathtt{A} - \lambda \mathtt{I}) \end{aligned}$$

e questo viene chiamato `polinomio caratteristico` della matrice $\mathtt{A}$. La soluzione di questo polinomio, le radici, forniscono gli autovalori di $\mathtt{A}$.

Teorema 5.0.1: Fondamentale dell'Algebra

$p_n(x)$ ha n radici complesse contate con la loro molteplicità.

Questo teorema ci assicura che $\exists$ n radici, il problema e che non esiste una formula risolutiva per polinomi di ordine superiore al sesto, quindi `funziona` solo per polinomi di grado basso, che si traduce in matrici di ordine n basso, il che ovviamente rende impraticabile, la determinazione degli autovalori percorrendo questa strada.

[1] Riprenderemo alcuni concetti visti nella sezione 2.2.1 a pagina 9.

5.1 Trasformazione di similitudine

Definizione 5.1.1: Similitudine

La matrice $\mathtt{B}$ è **simile** a $\mathtt{A}$ se $\exists \mathtt{X}$ non singolare : $\mathtt{X}^{-1}\mathtt{AX} = \mathtt{B}$ e si indica $\mathtt{A} \sim \mathtt{B}$

Teorema 5.1.1

$\mathtt{A} \sim \mathtt{B} \iff \sigma(\mathtt{A}) = \sigma(\mathtt{B})$

Dimostrazione. Teorema 5.1.1

$$\mathtt{Bx} = \lambda \mathtt{x} \Rightarrow \mathtt{X}^{-1}\mathtt{AXv} = \lambda \mathtt{v}$$
$$\mathtt{A}\underbrace{\mathtt{Xv}}_{\mathtt{w}} = \lambda \underbrace{\mathtt{Xv}}_{\mathtt{w}}$$
$$\mathtt{Aw} = \lambda \mathtt{w}$$

□

Definizione 5.1.2: Matrice diagonalizzabile

$$\mathtt{A} \sim \mathtt{D} = diag(\lambda_1, \lambda_2, \ldots, \lambda_n)$$
$$\exists \mathtt{X} : \mathtt{X}^{-1}\mathtt{DX} = \mathtt{D}$$
$$\mathtt{A} = \mathtt{XDX}^{-1} \quad \nexists \text{per tutte le matrici.}$$

Definizione 5.1.3: Matrice unitariamente diagonalizzabile

Se: $\mathtt{X} = \mathtt{U}$ dove $\mathtt{U}^*\mathtt{U} = \mathtt{UU}^* = \mathtt{I}$

$$\mathtt{U}^{-1}\mathtt{AU} = \mathtt{D} \Rightarrow \mathtt{U}^*\mathtt{AU} = \mathtt{D}$$

Si ricordi che mentre fare $\mathtt{X}^{-1}$ è un calcolo che porta inevitabilmente ad errori farne la trasposta (aggiunta) non **introduce errori**.

Teorema 5.1.2

$\mathtt{A}$ è diagonalizzabile $\iff$ gli autovettori sono **linearmente indipendenti**.

Dimostrazione. Teorema 5.1.2

$$\begin{aligned} \mathtt{X}^{-1}\mathtt{AX} = \mathtt{D}, \quad \mathtt{X} &= [\mathtt{v}_1|\mathtt{v}_2|\dots|\mathtt{v}_n] \\ \mathtt{AX} &= \mathtt{XD} \\ \mathtt{A}[\mathtt{v}_1|\mathtt{v}_2|\dots|\mathtt{v}_n] &= [\mathtt{v}_1|\mathtt{v}_2|\dots|\mathtt{v}_n]diag(\lambda_1, \lambda_2, \dots, \lambda_n) \\ \mathtt{X}^{-1}\mathtt{AX} &= \mathtt{D} \end{aligned}$$

$\mathtt{X}$ potrà essere invertita solo se le colonne sono linearmente indipendenti. □

Teorema 5.1.3: Schur

$$\mathtt{A} \in C^{n\times n} \exists \mathtt{U} \quad \mathtt{U}^*\mathtt{AU} = \mathtt{T} = \begin{bmatrix} \lambda_1 & * & \cdots & * \\ 0 & \lambda_2 & \cdots & * \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \lambda_n \end{bmatrix}$$

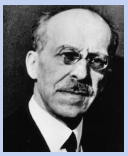
Corollario: Se $\mathtt{A} = \mathtt{A}^*$ (*Hermitiana*) $\Rightarrow$ è unitariamente diagonale.

$$\begin{gathered} \mathtt{A} = \mathtt{UTU}^* \quad \mathtt{A}^* = (\mathtt{UTU}^*)^* = \mathtt{UT}^*\mathtt{U}^* \Rightarrow \\ \mathtt{A} = \mathtt{A}^* \Rightarrow \mathtt{T} = \mathtt{T}^* \Rightarrow \mathtt{T} = \mathtt{D} \\ \lambda_i \in \mathbb{R} \ \mathtt{v}_i \perp \mathtt{v}_j \end{gathered}$$

gli autovalori sono reali e gli autovettori sono perpendicolari.

Molti problemi si concentrano sulla *localizzazione* che permette di sapere dove *non* sono gli autovalori. Altri algoritmi si concentrano sulla *separazione* degli autovalori. Un motore di ricerca utilizza questi algoritmi, si è interessati alla conoscenza dell'autovalore più grande in modulo, viceversa una rete sociale è interessata alla conoscenza dell'autovalore più piccolo (in modulo), un microblogging può essere interessato al penultimo autovalore etc.

Nota storica 5.1.1: Shur



Issai Schur (Mahilëŭ, 10 gennaio 1875 - Tel Aviv 10 gennaio 1941 (66 anni giusti giusti)) è stato un matematico tedesco. È conosciuto per i suoi lavori in rappresentazione dei gruppi, per quelli in combinatoria, teoria dei numeri e fisica teorica. [Wiki15b]

Cerchiamo l'autovalore più grande, possiamo raffinare la ricerca effettuando una deflazione la matrice *smontandola*, per esempio prendo una matrice 10×10, trovo l'autovalore più grande in modulo, elimino l'autovalore trovato ottenendo una matrice la matrice che diventa 9×9, ritrovo l'autovalore più grande e la matrice diventa 8×8, etc ...

Dall'autovalore posso trovare l'autovettore corrispondente, mentre se conosco l'autovettore posso trovare l'autovalore sfruttando il quoziente di Rayleigh:

$$\mathtt{Av} = \lambda\mathtt{v} \Rightarrow \frac{\mathtt{v}^T\mathtt{Av}}{\mathtt{v}^T\mathtt{v}} = \frac{\lambda\mathtt{v}^T\mathtt{v}}{\mathtt{v}^T\mathtt{v}} = \lambda \tag{5.1}$$

Teorema 5.1.4: Bauer-Fike

$$\mathtt{A} \in \mathbb{C}^{n\times n} \ \exists \mathtt{X} = \mathtt{X}^{-1}\mathtt{AX} = diag(\lambda_1, \lambda_2, \dots \lambda_n)$$
$$\mathtt{B} = \mathtt{A} + \mathtt{E}, \mu \in \sigma(\mathtt{B})$$
$$\min_{\lambda\in\sigma(\mathtt{A})} |\lambda - \mu| < k_2(\mathtt{X})\|\mathtt{E}\|_2.$$

dove la matrice $\mathtt{E}$ e la matrice degli errori, $k_2(\mathtt{X})$ è il numero di condizionamento della matrice: $\|\mathtt{X}\|_2\|\mathtt{X}^{-1}\|_2$ e μ è l'autovalore calcolato.

Corollario: Se $\mathtt{A} = \mathtt{A}^*$ $\mathtt{X}$è unitaria $\Rightarrow k_2(\mathtt{X}) = 1$
$\min_{\lambda 1 in\gamma(\mathtt{A})} |\lambda - \mu| \leq \|\mathtt{E}\|_2$. E' stabile quando gli autovettori sono perpendicolari: $\lambda_i \in \mathbb{R}$ $\mathtt{v}_i \perp \mathtt{v}_j$.

Nota storica 5.1.2: Bauer

Friedrich Ludwig "Fritz" Bauer (10 giugno 1924 - 26 Marzo 2015) è stato un informatico tedesco e professore. Bauer ha lavorato anche nei comitati che hanno sviluppato il linguaggio di programmazione *ALGOL*60. Nel 1968, Bauer ha coniato il termine `Software Engineering`. [Wiki17a].

5.2 Metodo iterativi

Consideriamo il sistema $\mathtt{Ax} = \mathtt{b}$, vogliamo risolverlo, possiamo usare dei metodi iterativi iniziamo a vederne uno: il Metodo delle potenze.

5.2.1 Metodo delle potenze

Consideriamo una matrice $\mathtt{A}$ $[n \times n]$ prendiamo un vettore a caso $\mathtt{x}^{(0)} \in \mathbb{R}^2$ e *costruiamo* un algoritmo:

$$\mathtt{x}^{(1)} = \mathtt{Ax}^{(0)}$$
$$\mathtt{x}^{(2)} = \mathtt{Ax}^{(1)}$$
$$\mathtt{x}^{(3)} = \mathtt{Ax}^{(2)}$$
$$\vdots$$
$$\mathtt{x}^{(n+1)} = \mathtt{Ax}^{(n)}$$

si verifica sempre che il vettore $\mathtt{x}^{(n)}$ tende a diventare l'autovettore, poiché in realtà l'autovettore indica una direzione. Prendendo spunto da questo algoritmo[3], possiamo svilupparne un altro, in particolare facendo delle ipotesi su $\mathtt{A}$:

1. diagonalizzabile: $\mathtt{A} = \mathtt{XDX}^{-1} \iff \mathtt{v}_i$ sono indipendenti;
2. $|\lambda_1| > |\lambda_i| \quad i = 2, 3, \dots, n$ sono ordinati i maniera decrescente;

[2]Per indicare che il procedimento è iterativo si mette un numero tra parentesi tonde all'esponente, le parentesi si mettono per considerare che non è un esponente.
[3]Che in genere crea problemi di under e over flow

3 $\mathtt{x}^{(0)}$ ha una componente non nulla nella $\mathtt{v}_1$.

Il primo vettore può essere scritto come un combinazione lineare di termini $\mathtt{x}^{(0)} = \sum_{i=1}^{n} \alpha_i \mathtt{v}_i$, pertanto:

$$\mathtt{x}^{(k)} = \mathtt{A}\mathtt{x}^{(k-1)} = \mathtt{A}\mathtt{A}\mathtt{x}^{(k-2)} = \mathtt{A}^2\mathtt{x}^{(k-2)} = \cdots = \mathtt{A}^k\mathtt{x}^{(0)}$$

$$\mathtt{x}^{(k)} = \mathtt{A}^k\mathtt{x}^{(0)} = \mathtt{A}^k\Big(\sum_{i=1}^{n} \alpha_i \mathtt{v}_i\Big) = \sum_{i=1}^{n} \alpha_i \mathtt{A}^k \mathtt{v}_i =$$

$$= \sum_{i=1}^{n} \alpha_i \lambda^k \mathtt{v}_i = \alpha_1 \lambda_1^k \mathtt{v}_1 + \sum_{i=2}^{n} \alpha_i \lambda_i^k \mathtt{v}_i$$

$$= \text{avendo sfuttato l'ipotesi } 3 \Rightarrow \alpha_1 \neq 0$$

$$= \alpha_1 \lambda_1^k \Big(\mathtt{v}_1 + \sum_{i=2}^{n} \frac{\alpha_i}{\alpha_1} \Big(\frac{\lambda_i}{\lambda_1}\Big)^k \mathtt{v}_i\Big)$$

poiché vale l'ipotesi 2 $|\frac{\lambda_i}{\lambda_1}| < 1$ e per $k \to \infty \Rightarrow |\frac{\lambda_i}{\lambda_1}| \to 0$

$$\mathtt{x}^{(k)} \to \alpha_1 \lambda_1^k \mathtt{v}_1$$

il limite tende ad essere un autovettore, il vettore: $\mathtt{x}^{(k)}$ tende ad essere parallelo a $\mathtt{v}_1$. Prendiamo in considerazione i seguenti casi:

$$\begin{aligned} &|\lambda_1| > 1 \quad \lambda_1^k \to \infty \quad \text{il vettore si allunga} \\ &|\lambda_1| < 1 \quad \lambda_1^k \to 0 \quad \text{il vettore sparisce} \\ &|\lambda_1| = 1 \quad \nexists \end{aligned}$$

come usare l'algoritmo? Ad ogni iterazione normalizzo $\mathtt{x}$ e *richiamo* il vettore:

$$\mathtt{q}^{(0)} = \frac{\mathtt{x}^{(0)}}{\|\mathtt{x}^{(0)}\|}$$

$$\vdots$$

$$\mathtt{q}^{(k+1)} = \frac{\mathtt{x}^{(k+1)}}{\|\mathtt{x}^{(k+1)}\|}$$

risolvo il sistema: $\mathtt{x}^{(k+1)} = \mathtt{A}\mathtt{q}^{(k)}$, ovvero trovo l'autovettore successivamente calcolo l'autovalore l'equazione 5.1 di Rayleigh, ricordando che $\|\mathtt{v}^T\mathtt{v}\| = \|\mathtt{q}^T\mathtt{q}\| = 1$:

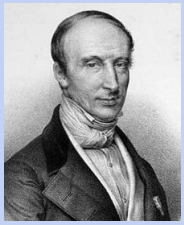
$$\lambda = \frac{\mathtt{v}^T\mathtt{A}\mathtt{v}}{\mathtt{v}^T\mathtt{v}} \Rightarrow$$

$$\lambda^{(k+1)} = \frac{\mathtt{q}^{(k+1)T}\mathtt{A}\mathtt{q}^{(k+1)}}{\|\mathtt{q}^{(k+1)T}\mathtt{q}^{(k+1)}\|}$$

$$= \frac{\mathtt{q}^{(k+1)T}\mathtt{A}\mathtt{q}^{(k+1)}}{1}$$

$$\lambda^{(k+1)} = \mathtt{q}^{(k+1)T}\mathtt{A}\mathtt{q}^{(k+1)}$$

Nota storica 5.2.1: Cauchy



Augustin-Louis Cauchy (Parigi, 21 agosto 1789 – Sceaux, 23 maggio 1857) è stato un matematico e ingegnere francese. Ha avviato il progetto della formulazione e dimostrazione rigorosa dei teoremi dell'analisi infinitesimale basato sull'utilizzo delle nozioni di limite e di continuità. [Wiki17b].

come posso terminare l'algoritmo? Fisso un limite. Criterio di Cauchy:

$$|\lambda^{(k+1)} - \lambda^{(k)}| < \tau$$

...e dal momento dal punto di vista numerico gli errori relativi sono *migliori* di quelli assoluti allora preferiamo:

$$\frac{|\lambda^{(k+1)} - \lambda^{k}|}{|\lambda^{(k+1)}|} < \tau \tag{5.2}$$

...e dal momento che la 5.2 non è proprio *buona* dal punto di vista numerico, preferiamo:

$$|\lambda^{(k+1)} - \lambda^{k}| < \tau|\lambda^{(k+1)}|. \tag{5.3}$$

Potrebbe accadere che la 3 nella pagina precedente non sia verificata e cioè: $\alpha_1 = 0$ questo comprometterebbe l'algoritmo ovvero potremmo trovare **altri** autovettori. Per **cercare** di evitare questo, prendiamo $\mathtt{x}^{(0)}$ assolutamente a caso. Questa è una condizione necessaria, ma non sufficiente, per evitare lo sfortunato caso in cui: $\alpha_1 = 0$.

Un'altra considerazione sulle ipotesi, se la 2 a pagina 30 non fosse verificata ovvero gli autovalori, in modulo, sono uguali, per esempio se accade che $\lambda_1 = 10; \lambda_2 = -10$ l'algoritmo **non converge**.

Algoritmo Metodo delle potenze

Si riporta l'algoritmo pronto per essere implementato.

Algoritmo: Metodo delle potenze
Input: $A, x^{(0)}, \tau, N_{max}$ *(numero massimo di iterazioni)*
Output: λ, q, k

1. $\mathtt{q}^{(0)} = \frac{\mathtt{x}^{(0)}}{\|\mathtt{x}^{(0)}\|}$
2. $k = 0$
3. $\lambda = \mathtt{q}^T\mathtt{A}\mathtt{q}$, flag = true
4. while flag
   - **a** $k = k + 1, \lambda_{old} = \lambda$
   - **b** $\mathtt{x} = \mathtt{A}\mathtt{q}$
   - **c** $\mathtt{q} = \frac{\mathtt{x}}{\|\mathtt{x}\|}$
   - **d** $\lambda = \mathtt{q}^T\mathtt{A}\mathtt{q}$
   - **e** flag $(|\lambda - \lambda_{old}| \geq \tau|\lambda|)$ AND $(k \leq N_{max})$

   end while

5.2.2 Eigenvector centrality

Questo algoritmo è la base di noto motore di ricerca, consideriamo la rete in figura 5.1, come possiamo considerare l'importanza di ciascun nodo? E come possiamo quantificarla? Scriviamo la matrice di adiacenza corrispondente alla rete, che risulta simmetrica poiché il grafo non è orientato.

$$\begin{bmatrix} 0 & 1 & 1 & 0 \\ 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} \quad \begin{cases} x_1 = x_2 + x_3 \\ x_2 = x_1 + x_3 + x_4 \\ x_3 = x_1 + x_2 \\ x_4 = x_2 \end{cases} \tag{5.4}$$

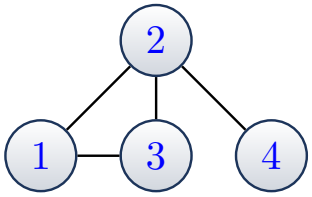


Figura 5.1: Rete

più in generale possiamo esprimere questa forma come $x_i = \frac{1}{\lambda}\sum_{j=1}^{n} a_{ij}x_j$ dove il fattore $\frac{1}{\lambda}$ è una costante, questa relazione può esprimersi come: $\mathtt{x} = \frac{1}{\lambda}\mathtt{Ax} \Rightarrow \mathtt{Ax} = \lambda\mathtt{x}$ dove x_i non può essere negativa.

Definizione 5.2.1: Matrice non negativa

Una matrice è non negativa se **tutti** gli elementi sono maggiori o uguali a 0.

Definizione 5.2.2: Primitiva

La matrice $\mathtt{A} \geq 0$ si dice primitiva se $\exists\, m > 0$: $\mathtt{A}^m \gg 0$.

Definizione 5.2.3: Riducibile

Una matrice quadrata $\mathtt{A}$ di dimensione n si dice **riducibile** se esiste una matrice di permutazione $\mathtt{P}$ tale che

$$\mathtt{PAP}^T = \begin{bmatrix} \mathtt{B}_{11} & \mathtt{B}_{12} \\ 0 & \mathtt{B}_{22} \end{bmatrix}$$

essendo $\mathtt{B}_{11}$ e $\mathtt{B}_{22}$ matici di dimensione $k \times k$ e $(n-k) \times (n-k)$, ripettivamente. Una matrice è **irriducibile** se non è riducibile.

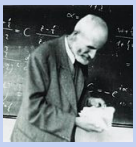
Teorema 5.2.1: Perron-Frobenius

Data una matrice quadrata $\mathtt{A}$, se questa è non negativa, primitiva e irriducibile allora:

1. L'autovalore λ di valore massimo in modulo è reale e positivo
2. λ è un autovalore semplice
3. l'autovettore corrispondente ha tutte le componenti positive
4. l'autovettore corrispondente è l'unico non negativo
5. $\rho(\mathtt{A})$ è una funzione strettamente crescente in ognuno dei suoi elementi.

Nota storica 5.2.2: Perron-Frobenius

Ferdinand Georg Frobenius (26 ottobre 1849 - 3 Agosto 1917) è stato un matematico tedesco, noto per i suoi contributi alla teoria delle funzioni ellittiche, equazioni differenziali e alla teoria dei gruppi. [Wiki17c]
Oskar Perron (7 maggio 1880 - 22 febbraio 1975) è stato un matematico tedesco. Ha scritto un libro enciclopedico sulle frazioni continue `Die Lehre von den Kettenbrüchen`. Ha introdotto il paradosso di Perron: Sia N il più grande intero. Se $N > 1$, allora $N^2 > N$, contraddice la definizione di N. Quindi $N = 1$.
Per illustrare il pericolo di supporre che la soluzione di un problema di ottimizzazione possa esiste. [Wiki15a].



Sfruttiamo il Metodo delle potenze e calcoliamoci, per la matrice 5.4, i pesi partendo da un vettore iniziale: $\mathbf{x}^{(0)} = [1, 1, 1, 1]^T$, allora il vettore $\mathbf{x}^{(1)} = [2, 3, 2, 1]^T$ etc. . . . giungeremo all'autovettore più grande.

Capitolo 6

Singular Value Decomposition

La scomposizione **SVD** riveste un ruolo fondamentale, non né verrà riportato l'algoritmo per la sua determinazione, ma si sfrutteranno direttamente i risultati. Prendiamo una **qualsiasi** matrice $\mathtt{A}$ essa può **sempre** essere decomposta come: $\mathtt{A} = \mathtt{U}\Sigma\mathtt{V}^T$, $\mathtt{A} \in \mathbb{R}^{m\times n}$; $\mathtt{U} = [\mathtt{u}_1|\mathtt{u}_2|\dots|\mathtt{u}_m]$; $\mathtt{V} = [\mathtt{v}_1|\mathtt{v}_2|\dots|\mathtt{v}_n]$; $\Sigma_{m\times n} = \begin{bmatrix} \sigma_1 & \cdots & 0 & \cdots & 0 \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ 0 & \cdots & \sigma_n & \cdots & 0 \end{bmatrix}$ dove: $\mathtt{U}^T\mathtt{U} = \mathtt{U}\mathtt{U}^T = \mathtt{I}_m, \mathtt{V}^T\mathtt{V} = \mathtt{V}\mathtt{V}^T = \mathtt{I}_n$,
$\sigma_i \in \mathbb{R}^+ \; \sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_n > 0, \quad k = rank(\mathtt{A})$.
Per un sistema lineare vale:

$$\mathtt{A}\mathtt{x} = [\mathtt{u}_1|\dots|\mathtt{u}_k|\mathtt{u}_{k+1}|\dots|\mathtt{u}_m] \begin{bmatrix} \sigma_1 & \cdots & 0 & \cdots & 0 \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ 0 & \cdots & \sigma_n & \cdots & 0 \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ 0 & \cdots & 0 & \cdots & 0 \end{bmatrix} \begin{bmatrix} \mathtt{v}_1^T \\ \vdots \\ \mathtt{v}_k^T \\ \mathtt{v}_{k+1}^T \\ \vdots \\ \mathtt{v}_n^T \end{bmatrix} \mathtt{x} =$$

$$= [\mathtt{u}_1|\dots|\mathtt{u}_k|\mathtt{u}_{k+1}|\dots|\mathtt{u}_m] \begin{bmatrix} \sigma_1 & \cdots & 0 & \cdots & 0 \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ 0 & \cdots & \sigma_n & \cdots & 0 \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ 0 & \cdots & 0 & \cdots & 0 \end{bmatrix} \begin{bmatrix} \mathtt{v}_1^T\mathtt{x} \\ \vdots \\ \mathtt{v}_k^T\mathtt{x} \\ \mathtt{v}_{k+1}^T\mathtt{x} \\ \vdots \\ \mathtt{v}_n^T\mathtt{x} \end{bmatrix} =$$

$$= [\mathtt{u}_1 \dots \mathtt{u}_k|\mathtt{u}_{k+1} \dots \mathtt{u}_m] \begin{bmatrix} \sigma_1\mathtt{v}_1^T\mathtt{x} \\ \vdots \\ \sigma_k\mathtt{v}_k^T\mathtt{x} \\ 0 \\ \vdots \\ 0 \end{bmatrix} = \sum_{i=1}^{k} \sigma_i(\mathtt{v}_i^T\mathtt{x})\mathtt{u}_i$$

da queste forme così come sono scritte si vedono i sottospazi vettoriali, in particolare:

- il nucleo: $\mathcal{N}(\mathtt{A}) = span[\mathtt{v}_{k+1} \dots \mathtt{v}_n]$

- il range: $\mathscr{R}(\mathtt{A}) = span[\mathtt{u}_1 \dots \mathtt{u}_k]$
- il nucleo trasposto: $\mathscr{N}(\mathtt{A}^*) = \mathscr{N}(\mathtt{A}^T) = span[\mathtt{u}_{k+1} \dots \mathtt{u}_n]$
- il range trasposto: $\mathscr{R}(\mathtt{A}^*) = \mathscr{R}(\mathtt{A}^T) = span[\mathtt{v}_1 \dots \mathtt{v}_k]$
- $\mathscr{N}(\mathtt{A})^\perp = \mathscr{R}(\mathtt{A}^T), \mathscr{R}(\mathtt{A})^\perp = \mathscr{N}(\mathtt{A}^T)$

Sistema omogeneo

Pensiamo di voler risolvere il problema lineare omogeneo, **immediatamente** possiamo saperne la soluzione, se conosciamo la scomposizione SVD, infatti

$$\mathtt{Ax} = \mathtt{0} \Rightarrow \mathscr{N}(\mathtt{A}) \neq \{\, 0 \,\} \Rightarrow \mathtt{x} = \sum_{i=k+1}^{n} \alpha_i \mathtt{v}_i$$

Norma

Vogliamo conoscere la norma 2:

$$\|\mathtt{A}\|_2 = \sqrt{\rho(\mathtt{A}\mathtt{A}^T)} = \sqrt{\rho(\mathtt{V}\Sigma^T\mathtt{U}^T\mathtt{U}\Sigma\mathtt{V}^T)} =$$
$$= \sqrt{\rho(\underbrace{\mathtt{V}\Sigma^2\mathtt{V}^T}_{\text{fatt.spett.}})} = \sigma_1$$

Numero di condizionamento

Il numero di condizionamento in norma 2, grazie alla scomposizione SVD può essere calcolato semplicemente:

$$k_2(\mathtt{A}) = \|\mathtt{A}\|_2\|\mathtt{A}^{-1}\|_2 = \frac{\sigma_1}{\sigma_n}$$

Prendendo come riferimento il teorema di Bauer-Fike 5.1.4 a pagina 30 possiamo generalizzarlo ad una matrice $m \times n$:

Teorema 6.0.1

$$\mathtt{A} \in \mathbb{C}^{m\times n}$$
$$|\sigma_i(\mathtt{A} + \mathtt{E}) - \sigma_i(\mathtt{A})| \leq \|\mathtt{E}\|_2. \tag{6.1}$$

Il problema è sempre stabile. La relazione 6 considera gli errori assoluti, questo potrebbe essere un problema se σ_i è piccolo, allora possiamo pensare di **scartare** i σ_i troppo piccoli e pensare di **troncare** la serie:

$$\mathtt{A} = \sum_{i=1}^{k} \sigma_i \mathtt{u}_i \mathtt{v}_i^T \approx \sum_{i=1}^{l} \sigma_i \mathtt{u}_i \mathtt{v}_i^T = \mathtt{A}_l.$$

Un teorema ci assicura la migliore approssimazione è data:

Teorema 6.0.2: Migliore approssimazione

$$arg \min_{rank(\mathtt{X})=l} \|\mathtt{A}-\mathtt{X}\|_2 = \mathtt{A}_l$$

quindi:

$$\|\mathtt{A}-\mathtt{A}_l\|_2 = \sigma_{l+1}$$
$$\|\mathtt{A}-\mathtt{A}_l\|_F = \sqrt{\sigma_{l+1}^2+\cdots+\sigma_k^2}$$

6.1 Un problema facile

Prendiamo un sistema lineare, con matrice $\mathtt{A} \in \mathbb{R}^{n\times n}$, potremmo usare Gauss, che essendo diretto è **corretto**, ma qua useremo il metodo iterativo SVD, pertanto:

$$\mathtt{Ax} = \mathtt{b} \Rightarrow \mathtt{U}\Sigma\mathtt{V}^T\mathtt{x} = \mathtt{b} \Rightarrow \Sigma \underbrace{\mathtt{V}^T\mathtt{x}}_{\mathtt{y}} = \underbrace{\mathtt{U}^T\mathtt{b}}_{\mathtt{c}}$$

$$\mathtt{c} = \mathtt{U}^T\mathtt{b} = \begin{bmatrix}\mathtt{u}_1^T\\ \mathtt{u}_2^T\\ \vdots\\ \mathtt{u}_n^T\end{bmatrix}\mathtt{b} = \begin{bmatrix}\mathtt{u}_1^T\mathtt{b}\\ \mathtt{u}_2^T\mathtt{b}\\ \vdots\\ \mathtt{u}_n^T\mathtt{b}\end{bmatrix}$$

$$\Sigma\mathtt{y} = \mathtt{c} \Rightarrow y_i = \frac{c_i}{\sigma_i} = \frac{\mathtt{u}_i^T\mathtt{b}}{\sigma_i}$$

$$\mathtt{y} = \mathtt{V}^T\mathtt{x} \Rightarrow \mathtt{Vy} = \mathtt{x} = [\mathtt{v}_1, \mathtt{v}_2, \cdots, \mathtt{v}_n]\begin{bmatrix}\mathtt{y}_1\\ \mathtt{y}_2\\ \vdots\\ \mathtt{y}_n\end{bmatrix} \Rightarrow \text{troviamo dunque il vettore } \mathtt{x}:$$

$$\mathtt{x} = \sum_{i=1}^{n} \frac{\mathtt{u}_i^T\mathtt{b}}{\sigma_i}\mathtt{v}_i \tag{6.2}$$

6.2 Un altro problemino . . . meno facile

Sicuramente ci sarà una matrice, che chiameremo con poca fantasia $\mathtt{A} \in \mathbb{R}^{m\times n}$: rettangolare cerchiamo di risolvere il problema dei minimi quadrati con minima norma: MNLSS (Minimum

Norm Least Square Solution) rappresentabile:

$$\begin{aligned} &\min \|\mathtt{x}\|_2 \\ s.a. &\min_{\mathtt{x}} \|\mathtt{Ax}-\mathtt{b}\|_2 \end{aligned} \tag{6.3}$$

data: $\mathtt{A} = \mathtt{U}\Sigma\mathtt{V}^T$

$$\begin{aligned} \|\mathtt{Ax}-\mathtt{b}\|_2 = \|\mathtt{U}\Sigma\mathtt{V}^T\mathtt{x}-\mathtt{b}\|_2 = \|\Sigma \underbrace{\mathtt{V}^T\mathtt{x}}_{\mathtt{y}} - \underbrace{\mathtt{U}^T\mathtt{b}}_{\mathtt{c}}\|_2 = \|\Sigma\mathtt{y}-\mathtt{c}\|_2 = \\ = \left\| \begin{bmatrix} \Sigma_k & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \mathtt{y}_1 \\ \mathtt{y}_2 \end{bmatrix} - \begin{bmatrix} \mathtt{c}_1 \\ \mathtt{c}_2 \end{bmatrix} \right\|_2 = \left\| \begin{bmatrix} \Sigma_k\mathtt{y}_1 \\ 0 \end{bmatrix} - \begin{bmatrix} \mathtt{c}_1 \\ \mathtt{c}_2 \end{bmatrix} \right\|_2 = \\ = \left\| \begin{bmatrix} \Sigma_k\mathtt{y}_1 - \mathtt{c}_1 \\ -\mathtt{c}_2 \end{bmatrix} \right\|_2 \end{aligned}$$

il vettore $\mathtt{c}_2$ è costante, posso minimizzare, annullandolo, il termine: $\Sigma_k\mathtt{y}_1 - \mathtt{c}_1 = 0 \Rightarrow \Sigma_k\mathtt{y}_1 = \mathtt{c}_1 \Rightarrow y_i = \frac{c_i}{\sigma_i}, i = 1 \dots k$. Scegliendo queste $\mathtt{y}$ annulliamo la prima riga, dunque il minimo risulta: $\|\mathtt{c}_2\|_2$, ma $\mathtt{y}$ può essere scritta:

$$\mathtt{y} = \mathtt{V}^T\mathtt{x} \Rightarrow \mathtt{x} = \mathtt{V}\mathtt{y} = \mathtt{V}\begin{bmatrix} \mathtt{y}_1 \\ \mathtt{y}_2 \end{bmatrix}$$

la seconda condizione delle 6.3 richiede $\min\|\mathtt{x}\|$ questo implica dato che ho ∞ soluzioni **devo** scegliere quella che minimizza $\mathtt{x}$ allora prendo $\mathtt{y}_2 = 0$, la soluzione normale. Allora:

$$\begin{aligned} \mathtt{x} = \mathtt{V}\begin{bmatrix} \mathtt{y}_1 \\ 0 \end{bmatrix} = [\mathtt{V}_1\mathtt{V}_2]\begin{bmatrix} \mathtt{y}_1 \\ 0 \end{bmatrix} = \mathtt{V}_1\mathtt{y}_1 = \\ = \sum_{i=1}^{k} y_i\mathtt{v}_i = \sum_{i=1}^{k} \frac{c_i}{\sigma_i}\mathtt{v}_i = \end{aligned}$$

$$\boxed{\mathtt{x} = \sum_{i=1}^{k} \frac{\mathtt{u}_i^T\mathtt{b}}{\sigma_i}\mathtt{v}_i} \tag{6.4}$$

la stessa dell'equazione 6.2.

6.3 Pseudo-inversa

Considerando la scomposizione SVD di una matrice $\mathtt{A} \in \mathbb{R}^{m\times n}$ vogliamo calcolare la pseudo-inversa:

$$\mathtt{x} = \mathtt{A}^\dagger\mathtt{b} = \mathtt{V}\mathtt{y} = \mathtt{V}\begin{bmatrix} \Sigma^{-1}\mathtt{c}_1 \\ 0 \end{bmatrix} = \mathtt{V}\begin{bmatrix} \frac{1}{\sigma_1} & \cdots & 0 & \cdots & 0 \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ 0 & \cdots & \frac{1}{\sigma_k} & \cdots & 0 \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ 0 & \cdots & 0 & \cdots & 0 \end{bmatrix}\begin{bmatrix} \mathtt{c}_1 \\ \mathtt{c}_2 \end{bmatrix} = \underbrace{\mathtt{V}\begin{bmatrix} \Sigma_k^{-1} & 0 \\ 0 & 0 \end{bmatrix}\mathtt{U}^T}_{\mathtt{A}^\dagger}\mathtt{b}$$

quindi possiamo riassumere come:

$$\mathtt{A} = \mathtt{U}\Sigma\mathtt{V}^T = \mathtt{U}\begin{bmatrix} \sigma_1 & \cdots & 0 & \cdots & 0 \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ 0 & \cdots & \sigma_k & \cdots & 0 \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ 0 & \cdots & 0 & \cdots & 0 \end{bmatrix}\mathtt{V}^T$$

$$\mathtt{A}^\dagger = \mathtt{V}\Sigma^\dagger\mathtt{U}^T = \mathtt{V}\begin{bmatrix} \frac{1}{\sigma_1} & \cdots & 0 & \cdots & 0 \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ 0 & \cdots & \frac{1}{\sigma_k} & \cdots & 0 \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ 0 & \cdots & 0 & \cdots & 0 \end{bmatrix}\mathtt{U}^T$$

Considerando ora una matrice $\mathtt{A} \in \mathbb{R}^{m\times n}$ **a rango pieno** $= n$:

$$\mathtt{A}^\dagger(\mathtt{A}^T\mathtt{A})^{-1}\mathtt{A}^T = (\mathtt{V}\Sigma^T \underbrace{\mathtt{U}^T\mathtt{U}}_{\mathtt{I}} \Sigma\mathtt{V}^T)^{-1}\mathtt{V}\Sigma^T\mathtt{U}^T = (\mathtt{V}\Sigma^2\mathtt{V}^T)^{-1}\mathtt{V}\Sigma^T\mathtt{U}^T =$$

$$= \mathtt{V}\Sigma^{-2}\underbrace{\mathtt{V}^T\mathtt{V}}_{\mathtt{I}}\Sigma^T\mathtt{U}^T = \mathtt{V}\begin{bmatrix} \frac{1}{\sigma_1^2} & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & \frac{1}{\sigma_n^2} \end{bmatrix}\begin{bmatrix} \sigma_1 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & \sigma_n \end{bmatrix}\mathtt{U}^T$$

Alcune proprietà della pseudo-inversa

1. $(\mathtt{A}^\dagger)^\dagger = \mathtt{A}$
2. $(\mathtt{A}^*)^T = (\mathtt{A}^T)^*$
3. $(\mathtt{A}^T\mathtt{A})^\dagger = \mathtt{A}^\dagger(\mathtt{A}^T)^\dagger$
4. $(\mathtt{AB})^\dagger \neq \mathtt{B}^\dagger\mathtt{A}^\dagger$
5. $\mathtt{A}\mathtt{A}^\dagger \neq \mathtt{A}^\dagger\mathtt{A}$

6.4 Usando la fattorizzazione QR

Teorema 6.4.1

Data una matrice $\mathtt{A} \in \mathbb{R}^{m \times n}$, non necessariamente a rango pieno, che per semplicità possiamo considerare con $m > n \Rightarrow \exists \mathtt{Q} : \mathtt{Q}\mathtt{Q}^T = \mathtt{Q}^T\mathtt{Q} = \mathtt{I}$

$$\mathtt{R} = \begin{bmatrix} \sigma_1 & * & * \\ \vdots & \ddots & * \\ 0 & \cdots & \sigma_n \\ 0 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & 0 \end{bmatrix} = \begin{bmatrix} \mathtt{R}_1 \\ \mathtt{0} \end{bmatrix}$$

$$\mathtt{A} = \mathtt{Q}\mathtt{R}$$

o in maniera compatta:

$$\mathtt{A} = \begin{bmatrix} \mathtt{Q}_1 \mathtt{Q}_2 \end{bmatrix} \begin{bmatrix} \mathtt{R}_1 \\ \mathtt{0} \end{bmatrix} = \mathtt{Q}_1 \mathtt{R}_1$$

$\mathtt{R}$ e $\mathtt{A}$ hanno la stessa SVD, infatti, se $\mathtt{R} = \mathtt{U}\Sigma\mathtt{V}^T$:

$$\mathtt{A} = \mathtt{Q}\mathtt{R} = \underbrace{(\mathtt{Q}\mathtt{U})}_{\widetilde{\mathtt{U}}} \Sigma \mathtt{V}^T = \widetilde{\mathtt{U}}\Sigma\mathtt{V}^T$$

ricordando che il prodotto tra due matrici ortogonali è una matrice ortogonale: $\widetilde{\mathtt{U}}\widetilde{\mathtt{U}}^T = \widetilde{\mathtt{U}}^T\widetilde{\mathtt{U}} = \mathtt{I}$

6.5 Un problema facile

Data $\mathtt{A} \in \mathbb{R}^{n \times n}$ non singolare, può essere decomposta: $\mathtt{A}\mathtt{x} = \mathtt{b} \Rightarrow \mathtt{Q}(\mathtt{R}\mathtt{x}) = \mathtt{b} \Rightarrow \mathtt{Q}\mathtt{y} = \mathtt{b}$

$$\begin{cases} \mathtt{Q}\mathtt{y} = \mathtt{b} \\ \mathtt{R}\mathtt{x} = \mathtt{y} \end{cases}$$

Usare la fattorizzazione QR per un sistema **rettangolare** è sicuramente preferibile rispetto a Gauss poiché la QR è più `stabile`, ma la preferiamo anche nel caso quadrato, come questo, per la propagazione degli errori, infatti:

1. **QR**: $k_2(\mathtt{R}) = k_2(\mathtt{A})$
2. **Gauss**: $k_\infty(\mathtt{U}) \leq 4^{n-1} k_\infty(\mathtt{A})$[1].

Usando LU si ha amplificazione degli errori, se non si usa il pivoting si ha ulteriore propagazione.

[1] Si potrebbe pensare che quanto detto sia inutile perché stiamo confrontando due numeri di condizionamento diversi, 2 per uno e ∞ per l'altro, è vero, ma il discorso rimane valido anche con questo confronto, si usa questo solo per una semplicità di calcolo.

6.6 Minimi quadrati

Consideriamo un problema, già visto, con la QR: la matrice $A \in \mathbb{R}^{m \times n}$ con $m > n$ e rango $= n$ (pieno), è noto che abbiamo una sola soluzione:

$$\min_x \|Ax - b\|_2^2 = \min_x \|QRx - b\|^2 = \min_x \|Rx - \underbrace{Q^T b}_{c}\|^2$$

$$\left\| \begin{bmatrix} R_1 \\ 0 \end{bmatrix} x - \begin{bmatrix} C_1 \\ C_2 \end{bmatrix} \right\|^2 = \left\| \begin{bmatrix} R_1 x - c_1 \\ -c_2 \end{bmatrix} \right\|^2 =$$

...e quindi scegliendo una x tale che $R_1 x = c_1$ allora il minimo vale $\|c_2\|^2$.

6.7 Sistema normale

Prendiamo la stessa matrice di sopra e la trasponiamo, abbiamo ∞ soluzioni, abbiamo un sistema del tipo: $A^T y = c$, dove possiamo scomporre la matrice A come: $A = QR$, e giungiamo al problema:

$$\begin{cases} \min \|y\| \\ \min \|A^T y - c\| \end{cases} \qquad \begin{cases} A^T A z = c \\ y = Az \end{cases}$$

svolgendo il prodotto delle matrici $A^T A$:

$$A^T A = R^T Q^T Q R = R^T R = [R_1^T \, 0] \begin{bmatrix} R_1 \\ 0 \end{bmatrix} = R_1^T R_1 \Rightarrow$$

$$R_1^T R_1 z \Rightarrow R_1 z = (R_1^T) c \Rightarrow y = Q \begin{bmatrix} R_1 \\ 0 \end{bmatrix} z = Q \begin{bmatrix} R_1 z \\ 0 \end{bmatrix} = Q \begin{bmatrix} (R_1^T)^{-1} c \\ 0 \end{bmatrix}$$

6.7.1 Caratteristiche degli algoritmi

Alcune Caratteristiche degli algoritmi:

1. **Gauss** $\mathcal{O}(\frac{1}{3}n^3)$ più veloce di QR;
2. **QR** $\mathcal{O}(\frac{2}{3}n^3)$ più stabile di Gauss.
3. **SVD** $\nexists$ è iterativo.

Capitolo 7

Metodi Iterativi per la risoluzione di sistemi lineari

I metodi iterativi per la risoluzione di un sistema lineare

$$\mathtt{A}\mathtt{x} = \mathtt{b} \tag{7.1}$$

generano, a partire da un vettore iniziale $\mathtt{x}^{(0)}$, una successione di vettori $\mathtt{x}^{(k)}, k = 0, 1, \ldots$, che, sotto opportune ipotesi, converge alla soluzione del problema. Nei metodi diretti che valutano la soluzione in un numero *finito* di passi, gli errori presenti nei risultati nascono esclusivamente dall'aritmetica finita e/o dalla presenza di errori sui dati. In quelli iterativi, invece, agli errori sperimentali e di arrotondamento si aggiungono gli errori di **troncamento**, derivati dal fatto che il limite cercato deve essere necessariamente approssimato troncando la successione per un indice sufficientemente grande. Si aggiunge, quindi, il problema della determinazione di un *criterio di arrotondamento*.

I metodi iterativi risultano convenienti per matrici di *grandi dimensioni*, specialmente se *strutturate* o *sparse*. Infatti, mentre un metodo diretto opera modificando la matrice de sistema, spesso alterandone la struttura e aumentando il numero degli elementi non nulli, un metodi iterativo, non richiede una modifica della matrice del sistema e in alcuni cassi neanche la sua memorizzazione. È solo necessario poter accedere in qualche modo ai suoi elementi.

Questo metodi possono, inoltre, essere utilizzati nei casi in cui si voglia raffinare una soluzione approssimata ottenuta con altri algoritmi o mediante informazioni a priori sul problema.

Infine è possibile ridurre il tempo di elaborazione, eseguendo un minor numero di iterazioni in quei casi in cui non sia richiesta un elevata accuratezza.

7.1 Metodi iterativi del prim'ordine

Definizione 7.1.1: Metodo iterativo del primo ordine

Il calcolo di un termine della successione coinvolge solo il termine precedente.

assume la forma

$$\mathtt{x}^{(k+1)} = \psi(\mathtt{x}^{(k)}).$$

Definizione 7.1.2: Globalmente convergente

Un metodo si dice globalmente convergente se per ogni vettore iniziale $\mathtt{x}^{(0)} \in \Re^n$ si ha che

$$\lim_{k\to\infty} \|\mathtt{x}^{(k)} - \mathtt{x}\| = 0,$$

dove $\mathtt{x}$ è la soluzione e $\|\cdot\|$ denota una qualsiasi norma.

Definizione 7.1.3: Consistente

Un metodo iterativo si dice **consistente** se

$$\mathtt{x}^{(k)} = \mathtt{x} \Rightarrow \mathtt{x}^{(k+1)} = \mathtt{x}.$$

Teorema 7.1.1

a consistenza è una condizione necessaria per la convergenza.

Un metodo **lineare, stazionario del primo ordine** assume la forma

$$\mathtt{x}^{(k+1)} = \mathtt{B}\mathtt{x}^{(k)} + \mathtt{f}. \tag{7.2}$$

La 7.2 è:

lineare perché è tale la relazione che la esprime;

stazionario perché la *matrice di iterazione* $\mathtt{B}$ e il vettore $\mathtt{f}$ non variano al variare dell'indice di iterazione k;

del prim'ordine perché il calcolo del vettore $\mathtt{x}^{(k+1)}$ dipende solo dal termine precedente $\mathtt{x}^{(k)}$.

Teorema 7.1.2

n metodo iterativo lineare, stazionario del prim'ordine è consistente se e solo se:

$$\mathtt{f} = (\mathtt{I} - \mathtt{B})\mathtt{A}^{-1}\mathtt{b}. \tag{7.3}$$

Definizione 7.1.4: Errore

Si definisce **errore** al passo k il vettore:

$$\mathtt{e}^{(k)} = \mathtt{x}^{(k)} - \mathtt{x}. \tag{7.4}$$

Teorema 7.1.3

ondizione sufficiente per la convergenza del metodo iterativo 7.2 è che esista una norma consistente $\|\cdot\|$ tale che $\|\mathtt{B}\| < 1$

Dimostrazione. Per le proprietà delle norme matriciali si ha che

$$\|\mathtt{e}^{(k)}\| \leq \|\mathtt{B}^k\| \, \|\mathtt{e}^{(0)}\| \leq \|\mathtt{B}\|^k \|\mathtt{e}^{(0)}\|,$$

quindi $\|\mathtt{B}\| < 1 \Rightarrow \|\mathtt{e}^{(k)}\| \to 0$. □

Teorema 7.1.4

n metodo iterativo lineare è convergente se e solo se il raggio spettrale $\rho(\mathtt{B})$ della matrice di iterazione $\mathtt{B}$ è minore di 1.

Dimostrazione. Sia $\rho(\mathtt{B}) < 1$, per ogni $\epsilon > 0$ $\exists$ una norma di matrice tale che

$$\|\mathtt{B}\| \leq \rho(\mathtt{B}) + \epsilon.$$

Essendo $\rho(\mathtt{B})$ strettamente minore a 1, è possibile scegliere ϵ in modo tale che $\rho(\mathtt{B}) + \epsilon < 1$. Il metodo iterativo risulta allora convergente. Se fosse $\rho(\mathtt{B}) \geq 1 \Rightarrow \exists$, un autovalore λ di $\mathtt{B}$ avente modulo maggiore o uguale a 1. Sia $\mathtt{v}$ l'autovettore corrispondente e scegliamo il vettore iniziale $\mathtt{x}^{(0)} = \mathtt{x} + \mathtt{v}$, essendo $\mathtt{x}$ la soluzione del sistema. In questo caso si avrebbe $\mathtt{e}^{(0)} = \mathtt{v}$ e quindi:

$$\begin{aligned}
\mathtt{B}\mathtt{e}^{(0)} &= \lambda \mathtt{e}^{(0)} \quad \Rightarrow \\
\mathtt{e}^{(k)} &= \mathtt{B}^k \mathtt{e}^{(0)} = \\
&= \mathtt{B}^{k-1}\mathtt{B}\mathtt{e}^{(0)} = \mathtt{B}^{k-1}\lambda \mathtt{e}^{(0)} = \\
&= \mathtt{B}^{k-2}\mathtt{B}\lambda \mathtt{e}^{(0)} = \mathtt{B}^{k-2}\lambda \mathtt{B}\mathtt{e}^{(0)} = \mathtt{B}^{k-2}\lambda^2 \mathtt{e}^{(0)} = \\
&\vdots \\
&= \lambda^k \mathtt{e}^{(0)}.
\end{aligned}$$

Poiché risulta $\|\mathtt{e}^{(k)}\| = |\lambda|^k \cdot \|\mathtt{e}^{(0)}\|$, nell'ipotesi che sia $|\lambda| \geq 1$ l'errore $\mathtt{e}^{(0)}$ non converge a zero. □

La convergenza è globale, quindi non dipende dal vettore iniziale $\mathtt{x}^{(0)}$.

7.2 Costruzione di metodi iterativi lineari

Una strategia spesso usata per la costruzione di metodi iterativi lineari è quella dello *splitting additivo*. Consiste nello scrivere la matrice del sistema nella forma:

$$\mathtt{A} = \mathtt{P} - \mathtt{N}, \tag{7.5}$$

dove la *matrice di precondizionamento* $\mathtt{P}$ è non singolare. Il sistema equivalente che si ottiene:

$$\mathtt{P}\mathtt{x} = \mathtt{N}\mathtt{x} + \mathtt{b},$$

che porta alla definizione del metodo iterativo

$$\mathtt{P}\mathtt{x}^{(k+1)} = \mathtt{N}\mathtt{x}^{(k)} + \mathtt{b}. \tag{7.6}$$

Il metodo costruito è consistente, e poiché $det(\mathtt{P}) \neq 0$ si può scrivere:

$$\mathtt{x}^{(k+1)} = \mathtt{P}^{-1}\mathtt{N}\mathtt{x}^{(k)} + \mathtt{P}^{-1}\mathtt{b} = \mathtt{B}\mathtt{x}^{(k)} + \mathtt{f},$$

ponendo $\mathtt{B} = \mathtt{P}^{-1}\mathtt{N}$ e $\mathtt{f} = \mathtt{P}^{-1}\mathtt{b}$. Per i Teoremi 7.1.3 e 7.1.4 una condizione di sufficienza per la convergenza è data da $\|\mathtt{P}^{-1}\mathtt{N}\| < 1$, dove $\|\cdot\|$ è una qualsiasi norma matriciale consistente, mentre una condizione necessaria e sufficiente è $\rho(\mathtt{P}^{-1}\mathtt{N}) < 1$. Perché il metodo risulti operativo è **necessario** che la matrice $\mathtt{P}$ sia più semplice da invertire di $\mathtt{A}$.

7.3 Criterio di arresto

Nell'applicazione di un metodo iterativo, risulta cruciale il criterio utilizzato per arrestare le iterazioni, in quanto da esso dipende la qualità dell'approssimazione della soluzione del sistema.

1. Un metodo per rilevare la convergenza consiste nel verificare che la successione delle iterate soddisfi il criterio di Cauchy, controllando lo scarto tra due iterazioni successive. Scegliendo:

   - una tolleranza $\tau > 0$;
   - norma vettoriale.

   La condizione di stop allora risulta:

   $$\|\mathtt{x}^{(k)} - \mathtt{x}^{(k-1)}\| \leq \tau, \tag{7.7}$$

   o meglio:

   $$\frac{\|\mathtt{x}^{(k)} - \mathtt{x}^{(k-1)}\|}{\|\mathtt{x}^{(k)}\|} \leq \tau, \tag{7.8}$$

   o meglio:

   $$\|\mathtt{x}^{(k)} - \mathtt{x}^{(k-1)}\| \leq \tau\|\mathtt{x}^{(k)}\|. \tag{7.9}$$

   Questa condizione non sarà mai verificata se il metodo non converge.

   Quando la matrice $\mathtt{A}$ è **simmetrica, definita positiva** può essere dimostrato che l'errore al passo k di un metodo iterativo del tipo 7.2 verifica la maggiorazione:

   $$\|\mathtt{e}^{(k)}\|_2 \leq \frac{1}{1-\rho(\mathtt{B})}\|\mathtt{x}^{(k+1)} - \mathtt{x}^{(k)}\|_2$$

   il che mostra che il test di arresto 7.7 è affidabile solo quando il raggio spettrale della matrice di iterazione $\mathtt{B}$ è sufficientemente inferiore a 1. Questo in generale è vero anche per una generica matrice $\mathtt{A}$.

   La scelta di τ deve essere ragionevole, dipende dalla precinzione richiesta nel problema in esame, ma che sia superiore alla precinzione di macchina.

2. Per considerare il caso in cui il metodo **non converga**, per evitare quell'effetto spiacevole di un *loop* ∞ è cosa buona e giusta fissare un limite massimo di iterazioni N.

3. Nei metodi iterativi per la risoluzione di sistemi lineari al passo k è possibile ottenere una maggiorazione per l'errore $\mathtt{e}^{(k)} = \mathtt{x}^{(k)} - \mathtt{x}$ in termini del *vettore residuo* $\mathtt{r}^{(k)} = \mathtt{b} - \mathtt{A}\mathtt{x}^{(k)}$.

   Può essere scritto:

$$\begin{aligned}\mathtt{A}\mathtt{e}^{(k)} &= \mathtt{A}\mathtt{x}^{(k)} - \mathtt{A}\mathtt{x} = \\ &= \mathtt{A}\mathtt{x}^{(k)} - \mathtt{b} = \\ &= -\mathtt{r}^{(k)},\end{aligned}$$

   da cui si ricava:

$$\|\mathtt{e}^{(k)}\| = \|\mathtt{A}^{-1}\mathtt{r}^{(k)}\| \leq \|\mathtt{A}^{-1}\| \, \|\mathtt{r}^{(k)}\|.$$

   Ricordando poi che $\|\mathtt{b}\| \leq \|\mathtt{A}\| \, \|\mathtt{x}\|$, si arriva alla seguente maggiorazione per l'errore relativo al passo k:

$$\frac{\|\mathtt{e}^{(k)}\|}{\|\mathtt{x}\|} \leq \|\mathtt{A}\| \, \|\mathtt{A}^{-1}\| \frac{\|\mathtt{r}^{(k)}\|}{\|\mathtt{b}\|} = \kappa(\mathtt{A}) \frac{\|\mathtt{r}^{(k)}\|}{\|\mathtt{b}\|}.$$

   Il criterio di stop:

$$\frac{\|\mathtt{r}^{(k)}\|}{\|\mathtt{b}\|} \leq \tau \tag{7.10}$$

   la stima dell'errore fornita dal residuo normalizzato è tanto attendibile quanto più la matrice $\mathtt{A}$ è mal condizionata.

In generale le condizioni di arresto per un criterio sono una combinazione di più criteri di arresto.

7.4 Precondizionamento

I metodi analizzati sono del tipo:

$$\mathtt{x}^{(k+1)} = \mathtt{B}\mathtt{x}^{(k)} + \mathtt{f}, \tag{7.11}$$

sfruttando la 7.5 a pagina 45 si ha:

$$\begin{aligned}\mathtt{B} &= \mathtt{P}^{-1}\mathtt{N} \\ &= \mathtt{P}^{-1}(\mathtt{P} - \mathtt{A}) \\ &= (\mathtt{I} - \mathtt{P}^{-1}\mathtt{A}), \\ \mathtt{f} &= \mathtt{P}^{-1}\mathtt{b}.\end{aligned}$$

Per la consistenza, ricordando la 7.1 e il teorema 7.1.2, questa relazione equivale al sistema lineare

$$(\mathtt{I} - \mathtt{B})\mathtt{x} = \mathtt{f},$$

ossia:

$$\begin{aligned}(\mathtt{I} - \mathtt{B})\mathtt{x} &= \mathtt{f} \\ (\mathtt{I} - \mathtt{P}^{-1}\mathtt{N})\mathtt{x} &= \mathtt{P}^{-1}\mathtt{b} \\ (\mathtt{I} - \mathtt{I} + \mathtt{P}^{-1}\mathtt{A})\mathtt{x} &= \mathtt{P}^{-1}\mathtt{b} \\ \mathtt{P}^{-1}\mathtt{A}\mathtt{x} &= \mathtt{P}^{-1}\mathtt{b}.\end{aligned}$$

Questo equivale a dire che il metodo iterativo risolve il sistema lineare **precondizionato** della matrice $\mathtt{P}$.

La scelta di un buon precondizionamento è un compito tutt'altro che facile, d'altronde sono possibile ∞ scelte. Per matrici di dimensione molto elevate si cerca di introdurre precondizionatori che preservino la struttura o la sparsità di $\mathtt{A}$.

Per accelerare la convergenza del metodo, è necessario fare in modo che $\mathtt{P}$:

1. approssimi la matrice $\mathtt{A}$, cosicché $\mathtt{P}^{-1}\mathtt{A} \simeq \mathtt{I}$,
2. facile da invertire.

Ricordano il teorema 7.1.4: è condizione necessaria e sufficiente per la convergenza del metodo iterativo 7.11 è

$$\rho(\mathtt{B}) = \rho(\mathtt{I} - \mathtt{P}^{-1}\mathtt{A}) < 1.$$

Indicando con μ_i e $\mathtt{v}_i$ gli autovalori e gli autovettori della matrice $\mathtt{P}^{-1}\mathtt{A}$ si ha che:

$$(\mathtt{I} - \mathtt{P}^{-1}\mathtt{A})\mathtt{v}_i = (1 - \mu_i)\mathtt{v}_i,$$

quindi la matrice di iterazione $\mathtt{B}$ ha autovalori $(1 - \mu_i)$ e autovettori $\mathtt{v}_i$. La condizione per convergenza diventa quindi:

$$\max_{i=1,\dots,n} |1 - \mu_i| < 1$$

cioè

$$0 < \mu_i < 2, \quad i = 1, \dots, n.$$

Dal momento che la convergenza sarà tanto più veloce quanto più il raggio spettrale di $\mathtt{B}$ è vicino a zero, questo significa che un buon precondizionatore deve far si che gli autovalori della matrice precondizionata $\mathtt{P}^{-1}\mathtt{A}$ presentino un **cluster** a 1, cioè che si accumulino in un intorno di 1.

I metodi lineari del prim'ordine possono essere espressi in funzione del residuo $\mathtt{r}^{(k)} = \mathtt{b} - \mathtt{A}\mathtt{x}^{(k)}$. Fissato lo *splitting* 7.5

$$\mathtt{x}^{(k+1)} = \mathtt{P}^{-1}\mathtt{N}\mathtt{x}^{(k)} + \mathtt{P}^{-1}\mathtt{b},$$

essendo $\mathtt{N} = \mathtt{P} - \mathtt{A}$,

$$\begin{aligned} \mathtt{x}^{(k+1)} &= (\mathtt{I} - \mathtt{P}^{-1}\mathtt{A})\mathtt{x}^{(k)} + \mathtt{P}^{-1}\mathtt{b} = \\ &= \mathtt{x}^{(k)} + \mathtt{P}^{-1}(-\mathtt{A}\mathtt{x}^{(k)} + \mathtt{b}) = \\ &= \mathtt{x}^{(k)} + \mathtt{P}^{-1}\mathtt{r}^{(k)}. \end{aligned} \tag{7.12}$$

Dalla 7.12 si vede che **se fosse** $\mathtt{P} = \mathtt{A}$ si avrebbe:

$$\mathtt{P}^{-1}\mathtt{r}^{(k)} = \mathtt{A}^{-1}(\mathtt{b} - \mathtt{A}\mathtt{x}^{(k)}) = \mathtt{x} - \mathtt{x}^{(k)}$$

la convergenza avverrebbe in un passo. Quando $\mathtt{P} \simeq \mathtt{A}$, il **residuo precondizionato** $\mathtt{P}^{-1}\mathtt{r}^{(k)}$ fornisce solo un'approssimazione dell'errore.

Per motivi numerici, il metodo viene applicato risolvendo il sistema:

$$\mathtt{P}\mathtt{z}^{(k)} = \mathtt{r}^{(k)},$$

senza invertire esplicitamente $\mathtt{P}$ e calcolando poi:

$$\mathtt{x}^{(k+1)} = \mathtt{x}^{(k)} + \mathtt{z}^{(k)}.$$

7.5 Iterazioni in sottospazi di Krylov

Nota storica 7.5.1: Krylov

Nikolay Mitrofanovich Krylov (29 Novembre 1879 San Pietroburgo – Maggio 11, 1955 Mosca) è stato un matematico russo noto per i suoi lavori sulla interpolazione, meccanica non-lineare e sulla fisica matematica. [Wiki17g].

Sia $A \in \mathbb{R}^n$. Per $r = 1, 2, \ldots,$ si definisce **sottospazio di Krylov** con indice r, generato da A e da $\mathtt{b}$, l'insieme:

$$\mathscr{K}_r := span(\mathtt{b}, \mathtt{Ab}, \ldots, \mathtt{A}^{r-1}\mathtt{b}) \subseteq \mathbb{R}^n.$$

I **metodi di proiezione in sottospazi di Krylov** determinano, per ogni r, una soluzione approssimata $\mathtt{x}^{(r)}$ del sistema lineare $\mathtt{Ax} = \mathtt{b}$ che appartenga al sottospazio $\mathscr{K}_r$. La scelta del vettore $\mathtt{x}^{(r)}$ può essere fatta applicando diversi criteri di ottimalità, che conducono a differenti algoritmi. Se i vettori $\mathtt{A}^k\mathtt{b}, k = 0, 1, \ldots, n-1$, risultassero tutti indipendenti si avrebbe

$$\mathscr{K}_n \equiv \mathbb{R}^n$$

e l'algoritmo terminerebbe in n passi. Se questa condizione non fosse verificata potrebbe avvenire che per un dato $l < n$ il vettore $\mathtt{A}^l\mathtt{b}$ risulti linearmente indipendente dai vettori precedenti. In questo caso esisterebbero scalari $\alpha_i, i = 0, \ldots, l$ non tutti nulli tali che

$$\sum_{i=0}^{l} \alpha_i \mathtt{A}^i\mathtt{b} = 0.$$

Supponendo $\alpha_0 \neq 0$, è possibile esplicitare in tale relazione il vettore $\mathtt{b}$, ottenendo:

$$\mathtt{Ax}^* = \mathtt{b} = -\frac{1}{\alpha_0}\sum_{i=1}^{l} \alpha_i \mathtt{A}^i\mathtt{b} = \mathtt{A}\left(-\frac{1}{\alpha_0}\sum_{i=1}^{l} \alpha_i \mathtt{A}^{i-1}\mathtt{b}\right),$$

dove $\mathtt{x}^*$ indica la soluzione esatta del sistema lineare, il che implica

$$\mathtt{x}^* = -\frac{1}{\alpha_0}\sum_{i=1}^{l} \alpha_i \mathtt{A}^{i-1}\mathtt{b} \in \mathscr{K}_l$$

il che implica che il metodo terminerebbe in l passi. Questi metodi terminando a n passi vengono comunque considerati iterativi (*pseudo-iterativi*) in quanto, se opportunamente precondizionati forniscono una soluzione sufficientemente precisa in un numero inferiore ad n.

7.5.1 Il gradiente coniugato come metodo di Krylov

Il metodo del gradiente coniugato è costituito da una iterazione in sottospazi di Krylov, come riportato dal teorema:

Teorema 7.5.1

ia $\mathtt{Ax} = \mathtt{b}$ con A simmetrica definita positiva. Se inizializzato con $\mathtt{x}^{(0)} = 0$, allora il metodo del gradiente coniugato procede fintantoché il residuo $\mathtt{r}^{(k)}$ non si annulla e valgono le seguenti identità per $l = 1, \ldots, k$

$$\begin{aligned}\mathscr{K}_l &:= span(\mathtt{b}, \mathtt{Ab}, \ldots, \mathtt{A}^{l-1}\mathtt{b}) = span(\mathtt{x}^{(1)}, \ldots, \mathtt{x}^{(l)}) \\ &= span(\mathtt{p}^{(0)}, \ldots, \mathtt{p}^{(l-1)}) = span(\mathtt{r}^{(0)}, \ldots, \mathtt{r}^{(l-1)}).\end{aligned}$$

Per ogni l si ha:

$$\mathtt{r}^{(l)T}\mathtt{r}^{(j)} = 0 \quad \text{e} \quad \mathtt{p}^{(l)T}\mathtt{A}\mathtt{p}^{(j)} = 0, \quad j = 0, 1, \ldots, l-1.$$

il teorema 7.5.1 mostra che il metodo del gradiente coniugato opera sostituendo la base $\mathscr{K}_l$ numericamente instabile

$$\left\{ \mathtt{b}, \mathtt{Ab}, \ldots, \mathtt{A}^{l-1}\mathtt{b} \right\}$$

con quella formata dalle direzioni A–coniugate $\left\{ \mathtt{p}^{(0)}, \ldots, \mathtt{p}^{(l-1)} \right\}$. La generica iterata $\mathtt{x}^{(l)}$ del metodo del gradiente coniugato ha una proprietà di ottimalità. Se la matrice A è simmetrica definita positiva la relazione

$$\|\mathtt{y}\|_A = \sqrt{\mathtt{y}^T\mathtt{A}\mathtt{y}}$$

definisce una norma matriciale di $\mathbb{R}^n$.

Teorema 7.5.2

l vettore $\mathtt{x}^{(l)}$ minimizza in $\mathscr{K}_l$ la norma–A dell'errore cioè

$$\mathtt{x}^{(l)} = arg \min_{\mathtt{x} \in \mathscr{K}_l} \|\mathtt{x}^* - \mathtt{x}\|_A$$

essendo $\mathtt{x}^*$ la soluzione esatta del sistema lineare. Inoltre se $\mathtt{e}^{(l)} = \mathtt{x}^* - \mathtt{x}^{(l)}$ allora per ogni l si ha

$$\|\mathtt{e}^{(l)}\|_A \leq \|\mathtt{e}^{(l-1)}\|_A$$

e si ottiene $\mathtt{e}^{(k)} = 0$ per ogni $k \leq n$.

7.5.2 L'iterazione di Arnoldi

Nota storica 7.5.2: Arnoldi

Walter Edwin Arnoldi (New York, Dicembre 14, 1917 - Ottobre 5, 1995) è stato un ingegnere meccanico Americano, principalmente noto per l'iterazione di Arnoldi, un algoritmo usato nell'algebra lineare numerica. [Wiki16a].

L'iterazione di Arnoldi consente di costruire una base $\{ \mathtt{q}_1, \ldots, \mathtt{q}_l \}$ per il sottospazio $\mathscr{K}_l$ formata da vettori tra loro ortonormali, cioè tali che $\mathtt{q}_i^T\mathtt{q}_j = \delta_{ij}$, sostituendo così la base numericamente instabile $\{ b \}, \mathtt{Ab}, \ldots, \mathtt{A}^{l-1}\mathtt{b}$. Ciò viene fatto applicando il metodo di ortogonalizzazione di

Gram-Schmidt:

$$\begin{aligned}
\mathtt{q}_1 &= \frac{\mathtt{b}}{\|\mathtt{b}\|},\\
\tilde{\mathtt{q}}_{s+1} &= \mathtt{A}^s\mathtt{b} - \sum_{i=1}^{s} h_{is}\mathtt{q}_i,\ s = 1,\dots,l-1\\
\mathtt{q}_{s+1} &= \frac{\tilde{\mathtt{q}}_{s+1}}{\|\tilde{\mathtt{q}}_{s+1}\|}
\end{aligned}$$

dove le norme vettoriali utilizzate sono euclidee e gli scalari h_{is} possono essere ricavati imponendo l'ortogonalità di $\tilde{\mathtt{q}}_{s+1}$ con i vettori precedentemente calcolati. Per motivi numerici, per la costruzione di $\tilde{\mathtt{q}}_{s+1}$ si utilizza il vettore $\mathtt{Aq}_s$ in luogo di $\mathtt{A}^s\mathtt{b}$. Infatti, pur trattandosi di due differenti vettori, questa scelta consente di passare da una base di $\mathscr{K}_s$ ad una di $\mathtt{K}_{s+1}$ mediante un algoritmo più stabile. Si giunge così alle seguenti formule, che consentono di costruire una base ortonormale per $\mathtt{K}_{l+1}$:

$$\begin{aligned}
\mathtt{q}_1 &= \frac{\mathtt{b}}{\|\mathtt{b}\|},\\
\tilde{\mathtt{q}}_{s+1} &= \mathtt{Aq}_s - \sum_{i=1}^{s} h_{is}\mathtt{q}_i,\ s = 1,\dots,l\\
\mathtt{q}_{s+1} &= \frac{\tilde{\mathtt{q}}_{s+1}}{h_{s+1,s}},\ s = 1,\dots,l,\\
h_{is} &= \mathtt{q}_i^T\mathtt{Aq}_s,\ i = 1,\dots,s,\\
h_{s+1,s} &= \|\tilde{\mathtt{q}}_{s+1}\|.
\end{aligned}$$

Di seguito l'Algoritmo:

Algoritmo: Iterazione di Arnoldi
Input: A, b, τ
Output:

1. $\mathtt{q}_1 = \mathtt{b}\|\mathtt{b}\|$
2. $s = 1$
3. repeat
   - $\mathtt{v} = \mathtt{Aq}_s$
   - for $i = 1,\dots,s$
     - ♣ $h_{is} = \mathtt{q}_i^T\mathtt{v}$
     - ♣ $\mathtt{v} = \mathtt{v} - h_{is}\mathtt{q}_i$
   - $h_{s+1,s} = \|\mathtt{v}\|$
   - $\mathtt{q}_{s+1} = \mathtt{v}/h_{s+1,s}$
   - $s = s + 1$
4. until $s = n + 1 OR |h_{s,s-1}| < \tau$

L'annullarsi, per un dato $l < n$, della norma $h_{l+1,l}$ interrompe l'algoritmo si parla allora di *breakdown*, e indica che il vettore $\mathtt{A}\mathtt{q}_l$ è **linearmente dipendente** dai precedenti e pertanto lo spazio $\mathscr{K}_l$ contiene la soluzione del sistema lineare $\mathtt{A}\mathtt{x} = \mathtt{b}$.

Una delle formule trovate può essere riscritta come:

$$\mathtt{A}\mathtt{q}_s = \sum_{i=1}^{s+1} h_{i,s}\mathtt{q}_i,\ s = 1, \dots, l,$$

esplicitando:

$$\begin{aligned}
s = 1 \qquad & \mathtt{A}\mathtt{q}_1 = \sum_{i=1}^{2} h_{i,s}\mathtt{q}_i = h_{1,1}\mathtt{q}_1 + h_{2,1}\mathtt{q}_2 \\
s = 2 \qquad & \mathtt{A}\mathtt{q}_2 = \sum_{i=1}^{3} h_{i,s}\mathtt{q}_i = h_{1,2}\mathtt{q}_1 + h_{2,2}\mathtt{q}_2 + h_{3,2}\mathtt{q}_2 \\
& \vdots \\
s = l \qquad & \mathtt{A}\mathtt{q}_l = \sum_{i=1}^{l+1} h_{i,s}\mathtt{q}_i = h_{1,l}\mathtt{q}_1 + h_{2,l}\mathtt{q}_2 + \cdots + h_{l+1,l}\mathtt{q}_{l+1}
\end{aligned}$$

si vede che può essere riscritta in maniera compatta:

$$\mathtt{A}\mathtt{Q}_l = \mathtt{Q}_{l+1}\tilde{\mathtt{H}}_l. \tag{7.13}$$

Dove $\mathtt{Q} = [\mathtt{q}_1, \mathtt{q}_2, \dots, \mathtt{q}_l] \in \mathbb{R}^{n\times l}$ e

$$\tilde{\mathtt{H}}_l = \begin{bmatrix}
h_{1,1} & h_{1,2} & \dots & \dots & h_{1,l} \\
h_{2,1} & h_{2,2} & \dots & \dots & h_{2,l} \\
 & h_{3,2} & \ddots & & h_{3,l} \\
 & & \ddots & \ddots & \vdots \\
 & & & h_{l,l-1} & h_{l,l} \\
 & & & & h_{l+1,l}
\end{bmatrix} \in \mathbb{R}^{(l+1)\times l}.$$

Dalla 7.13 si ottiene:

$$\mathtt{Q}_l^T\mathtt{A}\mathtt{Q} = \mathtt{H}_l := \begin{bmatrix}
h_{1,1} & h_{1,2} & \dots & \dots & h_{1,l} \\
h_{2,1} & h_{2,2} & \dots & \dots & h_{2,l} \\
 & h_{3,2} & \ddots & & h_{3,l} \\
 & & \ddots & \ddots & \vdots \\
 & & & h_{l,l-1} & h_{l,l}
\end{bmatrix} \in \mathbb{R}^{(l)\times l}.$$

Che per $l = n$ diventa:

$$\mathtt{Q}_n^T\mathtt{A}\mathtt{Q}_n = \mathtt{H}_n,$$

l'iterazione di Arnoldi porta alla matrice di Hessemberg mediante n trasformazioni ortogonali. Essendo le matrice $\mathtt{A}$ e $\mathtt{H}_n$ unitariamente simili, questo potrebbe essere visto come un passo preliminare all'applicazione di un metodo per la ricerca degli autovalori che tragga vantaggio dalla struttura di Hessemberg, come l'algoritmo QR.

in realtà la matrice $\mathtt{H}_l$ contiene importanti informazioni sulla matrice $\mathtt{A}$ anche per l molto minore di n. In molti casi gli autovalori estremali di $\mathtt{H}_l$ convergono rapidamente agli autovalori estremali di $\mathtt{A}$ e continuando con le iterazioni un numero sempre maggiore di autovalori delle due matrici tende a coincidere. Questo consente, nel caso di matrici di grandi dimensioni di ottenere dopo *poche* iterazioni approssimazioni sufficientemente accurate degli autovalori estremali dalla matrice in esame, i quali spesso consentono di determinare importanti informazioni sulla matrice stessa (raggio spettrale, condizionamento, etc.) o sul problema fisico da cui la matrice deriva.

7.5.3 L'iterazione di Lanczos

Nota storica 7.5.3: Lanczos

Cornelius (Cornel) Lanczos era un matematico e fisico ungherese ebreo, nato il 2 febbraio 1893 (Székesfehérvár) e morto il 25 giugno 1974 (Budapest). [Wiki17h].

Supponendo di applicare l'iterazione di Arnoldi ad una matrice $\mathtt{A}$ simmetrica. Per questo particolare caso si ricava, che la matrice $\mathtt{H}_l = \mathtt{Q}_l^T \mathtt{A} \mathtt{Q}_l$, detta matrice di *Ritz*, dovendo essere simultaneamente simmetrica e in forma di Hessemberg, deve necessariamente essere tridiagonale. Essa sarà tale quindi che i suoi autovalori, i *valori di Ritz*, sono reali. Il metodo di Lanczos consente di ottenere una fattorizzazione $\mathtt{A}\mathtt{Q}_l = \mathtt{Q}_{l+1}\tilde{\mathtt{H}}_l$ dove $\mathtt{Q}_l = [\mathtt{q}_1, \ldots, \mathtt{q}_l] \in \mathbb{R}^{n \times l}$ è costituita da colonne ortonormali che costituiscono una base per il sottospazio di Krylov di dimensione l.

Ponendo $h_{i,i} = \alpha_i$ e $h_{i+1,i} = h_{i,i+1} = \beta_i$ possiamo scrivere la matrice $\mathtt{H}_l$:

$$\mathtt{H}_l = \begin{bmatrix} \alpha_1 & \beta_1 & & \\ \beta_1 & \alpha_2 & \ddots & \\ & \ddots & \ddots & \beta_{l-1} \\ & & \beta_{l-1} & \alpha_l \end{bmatrix}$$

Lo sfruttamento della particolare struttura della matrice porta ad un abbattimento della complessità computazionale. Il metodo, che prende il nome di **iterazione di Lanczos**, assume la forma:

$$\begin{aligned} \mathtt{q}_1 &= \frac{\mathtt{b}}{\|\mathtt{b}\|}, \\ \tilde{\mathtt{q}}_{s+1} &= \mathtt{A}\mathtt{q}_s - \alpha_s \mathtt{q}_s - \beta_{s-1}\mathtt{q}_{s-1}, \ s = 1, \ldots, l \\ \mathtt{q}_{s+1} &= \frac{\tilde{\mathtt{q}}_{s+1}}{\beta_s}, \ s = 1, \ldots, l, \end{aligned}$$

con

$$\begin{aligned} \alpha_s &= \mathtt{q}_s^T \mathtt{A} \mathtt{q}_s, \\ \beta_s &= \|\tilde{\mathtt{q}}_{s+1}\|. \end{aligned}$$

Nell'algoritmo, se si dovesse verificare, per un certo passo $s < n$ un *breakdown* in output avrà le matrici $\mathtt{Q}$ e $\mathtt{H}$ ridotte e cioè di dimensioni $n \times s$ e $s \times s$ rispettivamente, dove s è l'indice dove si verifica il breakdown.

Il presentarsi del breakdown nel caso dell'applicazione del metodo alla risoluzione di un sistema lineare $\mathtt{Ax} = \mathtt{b}$ implica che il sottospazio di Krylov $\mathscr{K}_l$ conterrà la soluzione esatta del sistema lineare simmetrico.

Algoritmo: Iterazione di Lanczos
Input: A, b, τ
Output:

1. $\beta_0 = 0, \mathtt{q}_0 = 0$
2. $\mathtt{q}_1 = \mathtt{b}/\|\mathtt{b}\|$
3. $s = 1$
4. repeat
   - $\mathtt{v} = \mathtt{A}\mathtt{q}_s$
   - $\alpha_s = \mathtt{q}_s^T \mathtt{v}$
   - $\mathtt{v} = \mathtt{v} - \alpha_s \mathtt{q}_s - \beta_{s-1}\mathtt{q}_{s-1}$
   - $\beta_s = \|\mathtt{v}\|$
   - $\mathtt{q}_{s+1} = \mathtt{v}/\beta_s$
   - $s = s + 1$
5. until $s = n + 1 OR |h_{s,s-1}| < \tau$

7.5.4 Il metodo GMRES

Dato un sistema lineare:

$$\mathtt{Ax} = \mathtt{b},$$

con $\mathtt{A}$ non singolare. I metodi GMRES (*Generalized Minimal RESidual*) consiste nel costruire iterativamente una base ortonormale per lo spazio di Krylov $\mathscr{K}_l$, per $l = 1, \ldots, n$, approssimando, ad ogni iterazione, la soluzione del sistema $\mathtt{x}^*$ col vettore $\mathtt{x}^{(l)} \in \mathscr{K}_l$ che minimizza la norma−2 del residuo: $\mathtt{r}^{(l)} = \mathtt{b} - \mathtt{A}\mathtt{x}^{(l)}$.

Ponendo

$$\mathtt{x}^{(l)} = arg \min_{x \in \mathscr{K}_l} \|\mathtt{b} - \mathtt{Ax}\|_2 \tag{7.14}$$

proiettiamo la soluzione nello spazio di Krylov $\mathscr{K}_l$ e la valutiamo risolvendo un problema ai minimi quadrati. Pur trattandosi si un metodo che termina la più in n passi, le proprietà dell'iterazione di Arnoldi fanno si che in molti casi si ottenga un'approssimazione sufficientemente accurata della soluzione del sistema in un numero di passi molto inferiore a n.

Un qualunque vettore di $\mathscr{K}_l$ può essere espresso nella base ortonormale $\{ \mathtt{q}_1, \ldots, \mathtt{q}_l \}$ nella forma $\mathtt{Q}_l\mathtt{y}$, essendo $\mathtt{y}$ un generico vettore di $\mathbb{R}^l$. Dunque

$$\begin{aligned}\|\mathtt{b} - \mathtt{A}\mathtt{x}\| &= \|\mathtt{A}\mathtt{Q}_l\mathtt{y} - \mathtt{b}\| = \\ &= \|\mathtt{Q}_{l+1}\tilde{\mathtt{H}}_l\mathtt{y} - \mathtt{b}\| = \\ &= \|\mathtt{Q}_{l+1}\tilde{\mathtt{H}}_l\mathtt{y} - \mathtt{Q}_{l+1}\mathtt{Q}_{l+1}^T\mathtt{b}\| = \\ &= \|\mathtt{Q}_{l+1}(\tilde{\mathtt{H}}_l\mathtt{y} - \mathtt{Q}_{l+1}^T\mathtt{b})\| = \\ &= \|\tilde{\mathtt{H}}_l\mathtt{y} - \|b\|\mathtt{e}_1\|.\end{aligned}$$

Avendo sfruttato la 7.13 e ricordando che $\mathtt{Q}_{l+1}^T\mathtt{Q}_{l+1} = \mathtt{I}_{l+1}$, $\mathtt{Q}_{l+1}^T\mathtt{b} = \|b\|\mathtt{e}_1$ dove $\mathtt{e}_1$ il vettore 1 della base canonica: $\mathtt{e}_1 = [1, 0, \ldots, 0]^T$.

Per il problema 7.14 è possibile calcolare la soluzione $\mathtt{y}^{(l)}$ di:

$$\min_{\mathtt{y}\in\mathbb{R}^l} = \|\tilde{\mathtt{H}}_l\mathtt{y} - \|b\|\mathtt{e}_1\|,$$

ponendo successivamente $\mathtt{x}^{(l)} = \mathtt{Q}\mathtt{y}^{(l)}$. Questo problema di minimo oltre a essere non vincolato, è caratterizzato da una matrice dei coefficienti in $\mathbb{R}^{(l+1)\times l}$, quindi di dimensioni contenute fintantoché l'indice di iterazione non cresce. La forma di Hessemberg rende agevole la soluzione del problema ai minimi quadrati.

Capitolo 8

Risultati ottenuti

Le simulazioni svolte sono centrate sulla risoluzione dei sistemi lineari equaione 7.1, per ciascuna si procede in tal modo:

1. si prende una matrice `A` random,
2. si fissa il vettore soluzione $\hat{\mathtt{x}}$ come vettore di tutti *uno*,
3. si *costruisce* il vettore di termini noti $\mathtt{b} = \mathtt{A}\hat{\mathtt{x}}$,
4. si risolve il sistema lineare: $\mathtt{Ax} = \mathtt{b}$, si trova la `x` (che è diversa da $\hat{\mathtt{x}}$),
5. si calcola l'errore: $\|\mathtt{x} - \hat{\mathtt{x}}\|$,
6. si calcola anche il tempo di calcolo con in comandi: `tic, toc`,
7. si confrontano tempi e errori tra gli algoritmi presentati e gli algoritmi di Matlab® .

Si riportano anche i risultati sulla fattorizzazione QR.

`Ogni simulazione` può essere divisa in 3 `fasi` distinte: preprocessor, processor, postprocessor.

La prima è quella che comprende i dati iniziali, gli input: la dimensione massima della matrice, i margini di errore che si vuole commettere etc.

La seconda: la vera simulazione, nel codice qui sono sicuramente presenti: gli algoritmi veri e propri, e sicuramente le tic, toc utilizzate per calcolare i tempi.

La terza scrittura su disco i risultati ottenuti esportazione dei risultati in file `leggibile` da LaTeX, per l'inportazione.

8.1 Le simulazioni fattorizzazione Cholesky

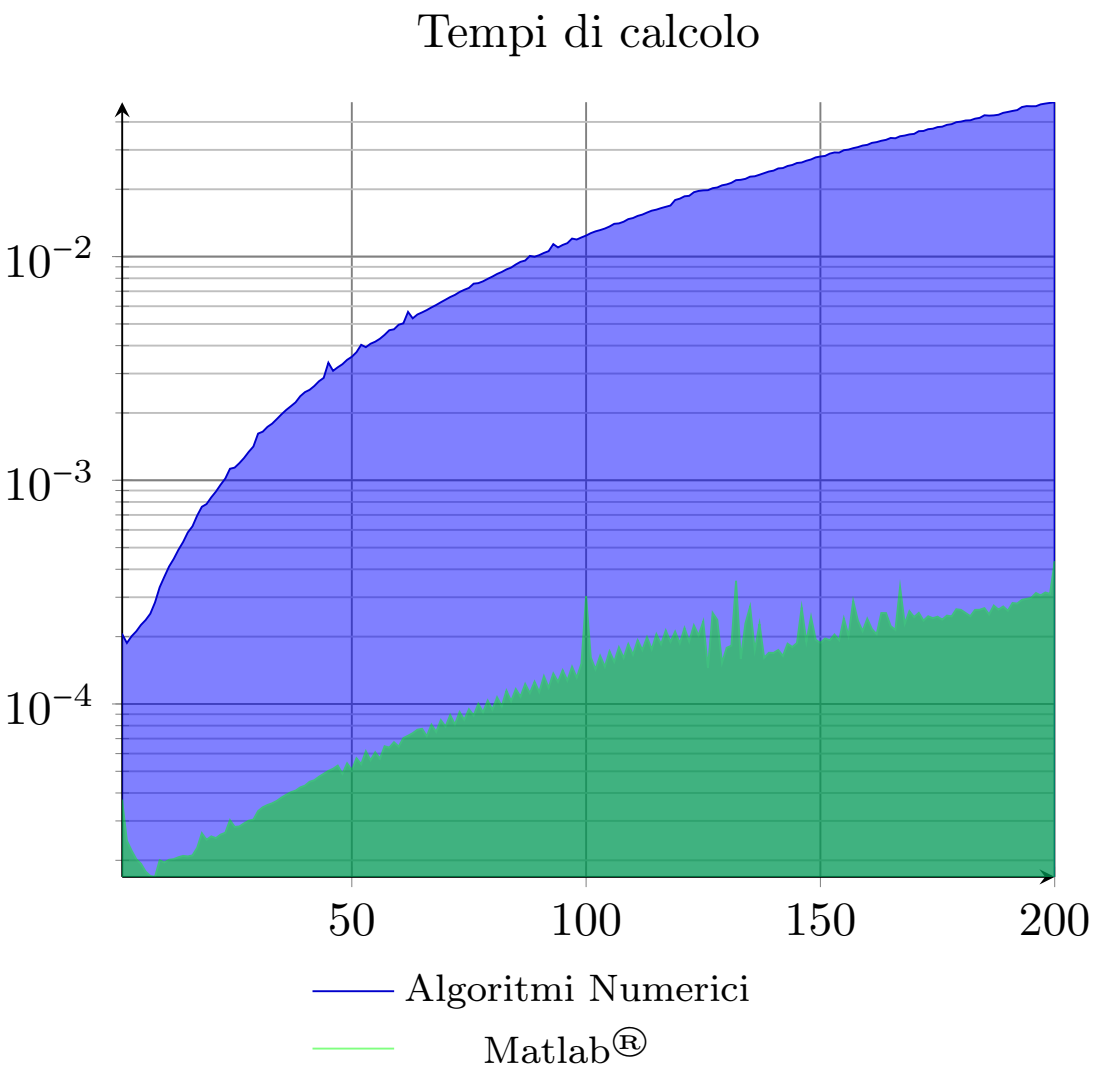


Figura 8.1: Fattorizzazione Cholesky per colonne. Vengono confrontati i tempi di calcolo di due Algoritmi, l'Algoritmo implementato e l'Algoritmo `chol()` di Matlab® . Si risolve il sistem lineare, cosiderando una matrice piena casuale random, si risolvono così due sistemi uno triangoalre inferiore e uno triangolare superiore, Nel grafico in acscisse è riportata la dimensione della matrice da 3 a 200, in ordinate in scala logaritmica i tempi di calcolo in [s].

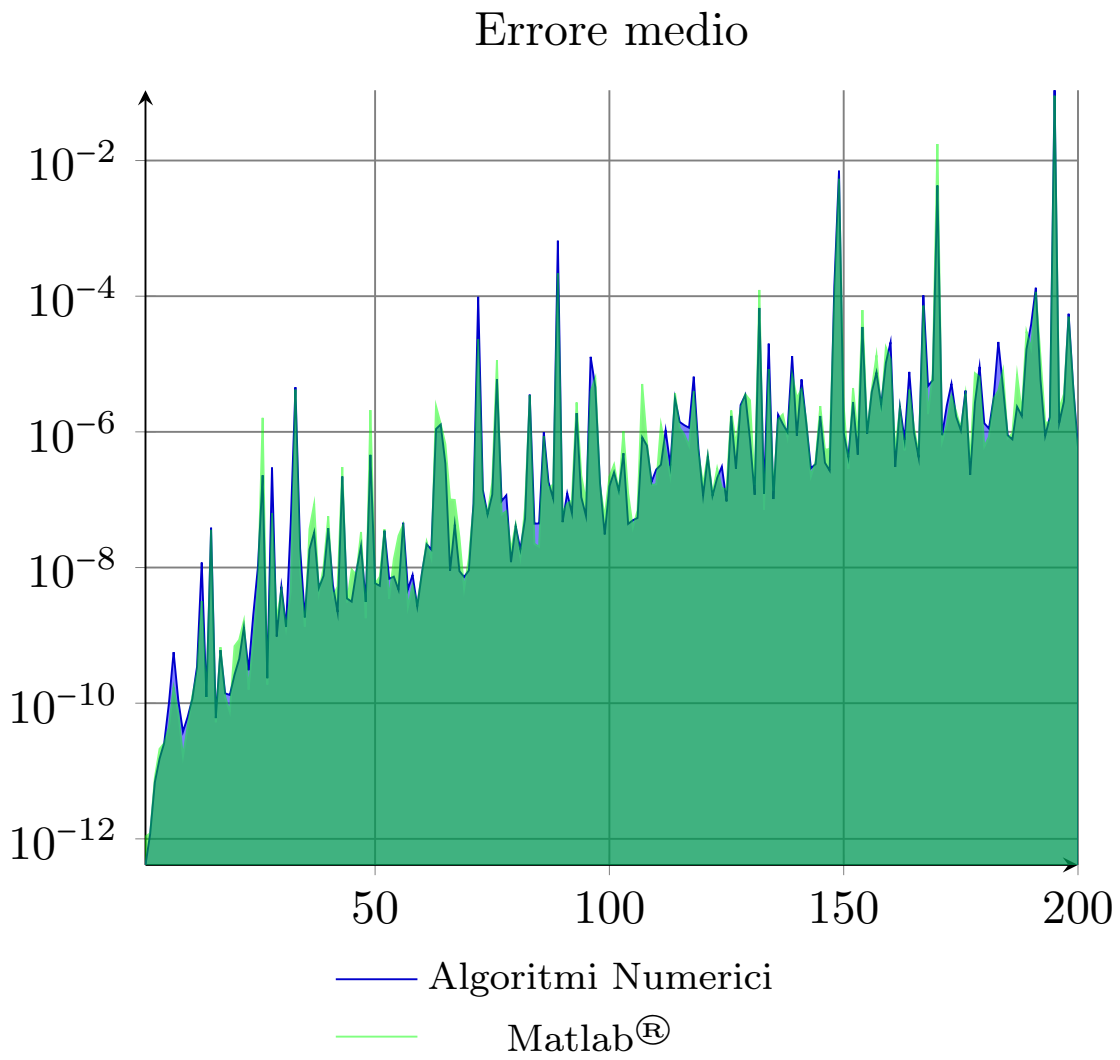


Figura 8.2: Fattorizzazione Cholesky. In ascisse sono riportate le dimensioni della matrice `A` da 3 a 200, in ordinate, in scala logaritmica è riportata la norma dell'errore calcolato come differenza tra soluzione calcolata e soluzione esatta.

8.2 Le simulazioni fattorizzazione QR

Prima di procedere con la sperimentazione sul calcolo dei minini quadrati con fattorizzazione QR si sono anlizzate e confrontate le fattorizzazioni QR: Householder, Givens, CGD, MGS, Matlab® . Si riportano le singole fattorizzazioni confrontate **tutte** con `qr()`. In tutte le simulazioni le matrici sono sempre prese quadrate.

8.2.1 Householder

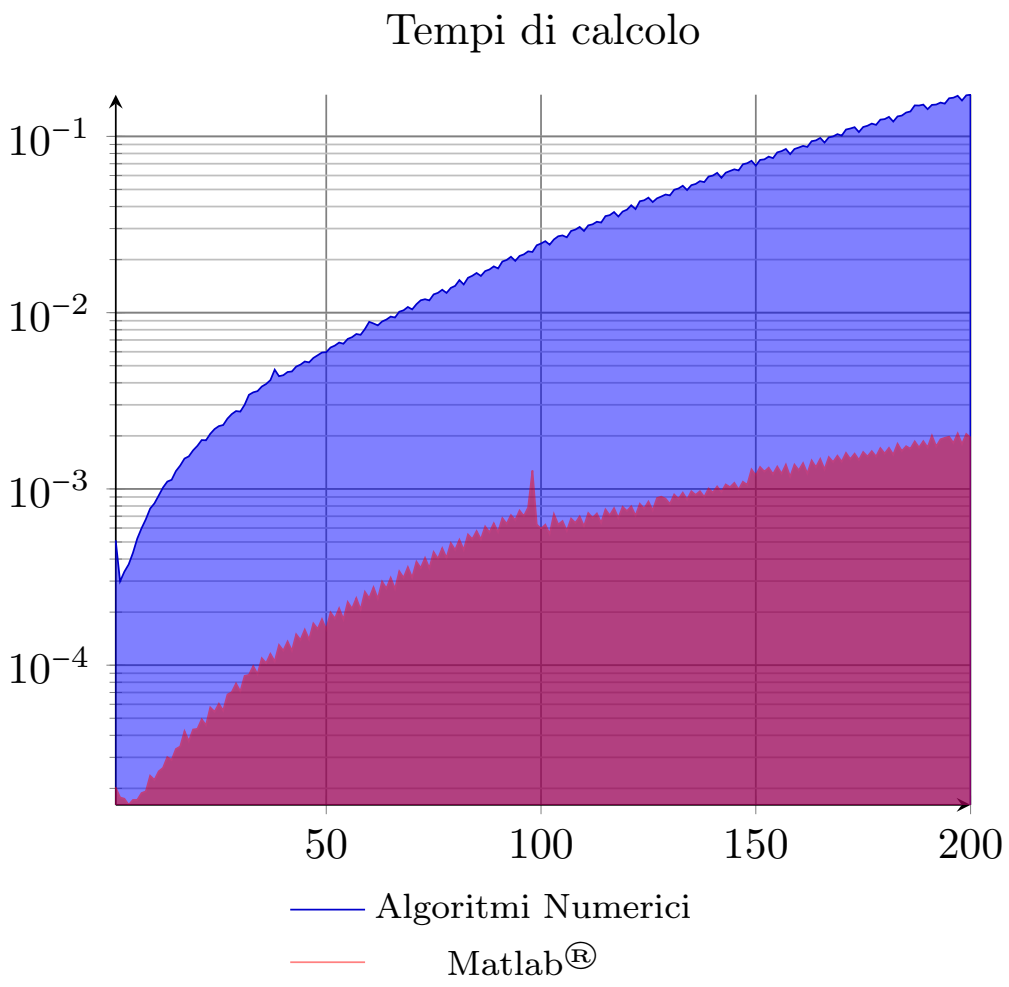


Figura 8.3: Fattorizzazione QR di Householder ($m = n$). Vengono confrontati i tempi di calcolo di due Algoritmi, l'Algoritmo è quello di pagina 102 libro di testo [Rod08] e l'Algoritmo `qr()` di Matlab® .

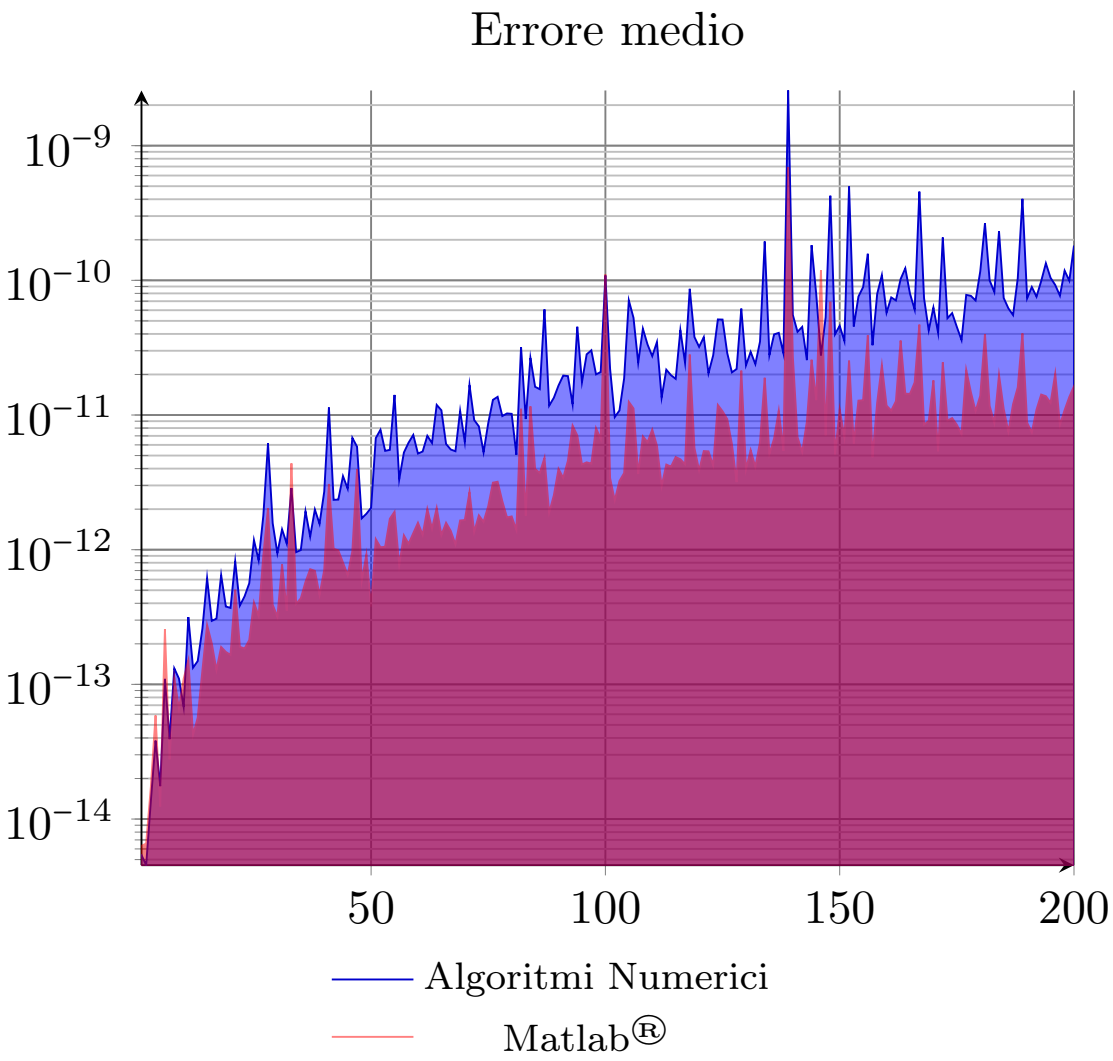


Figura 8.4: Fattorizzazione QR di Householder ($m = n$). Errori commessi.

8.2.2 Givens

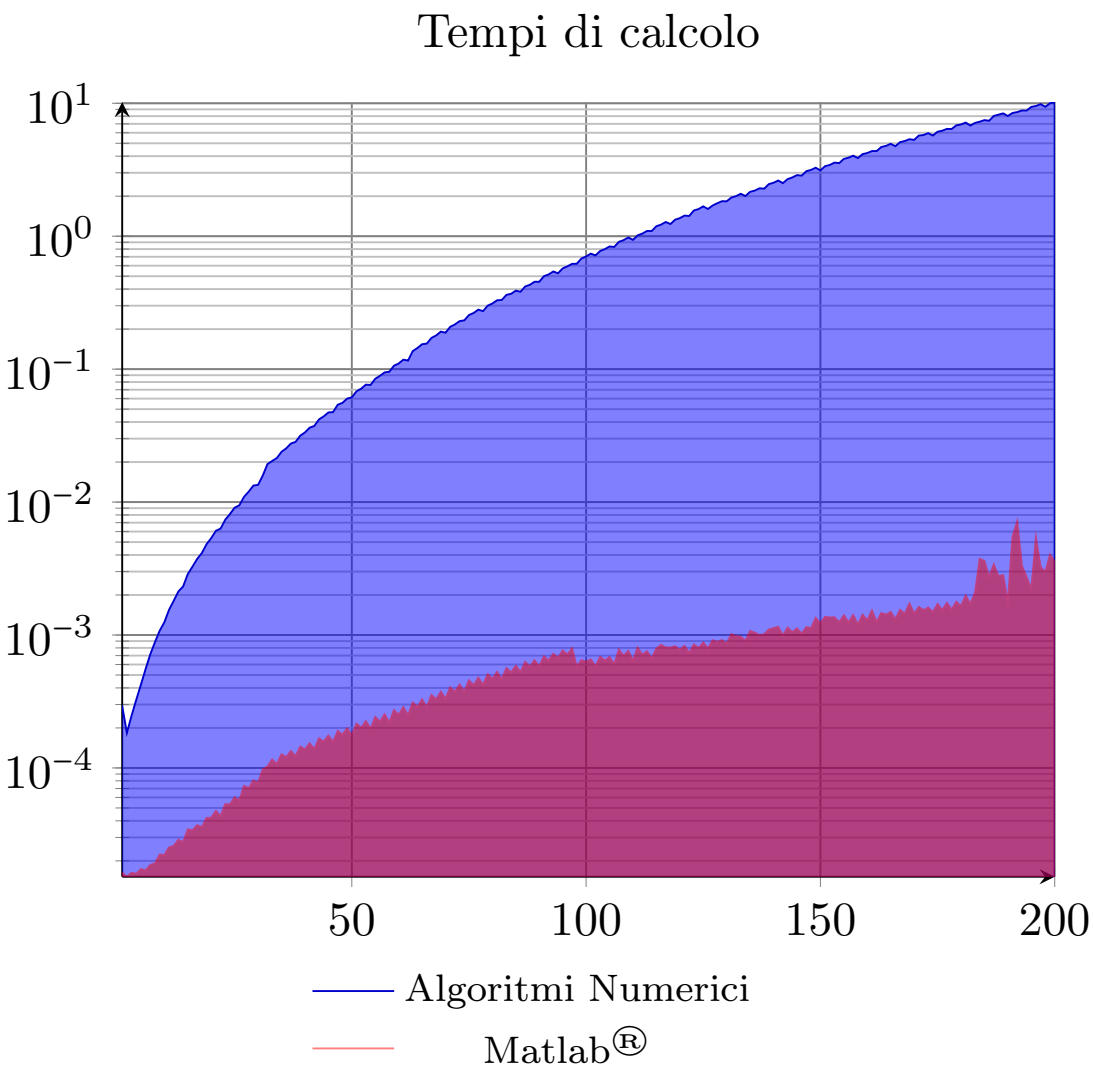


Figura 8.5: Fattorizzazione QR di Givens ($m = n$). Tempi di calcolo.

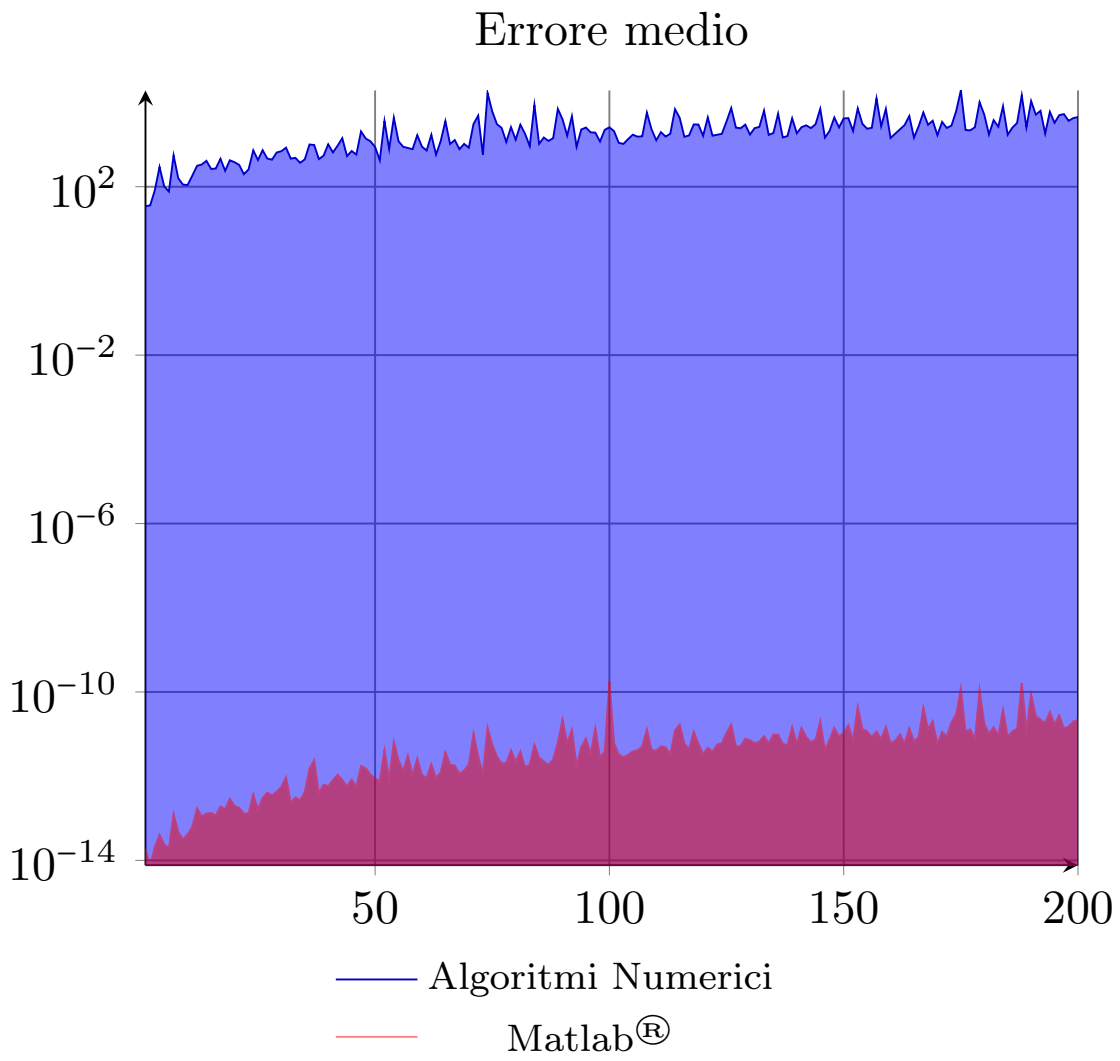


Figura 8.6: Fattorizzazione QR di Givens ($m = n$). Errori di calcolo.

8.2.3 Gram-Schmidt

Per questo caso,ma solo per questo caso, si è simulato **solo** classic Gram-Schmidt.

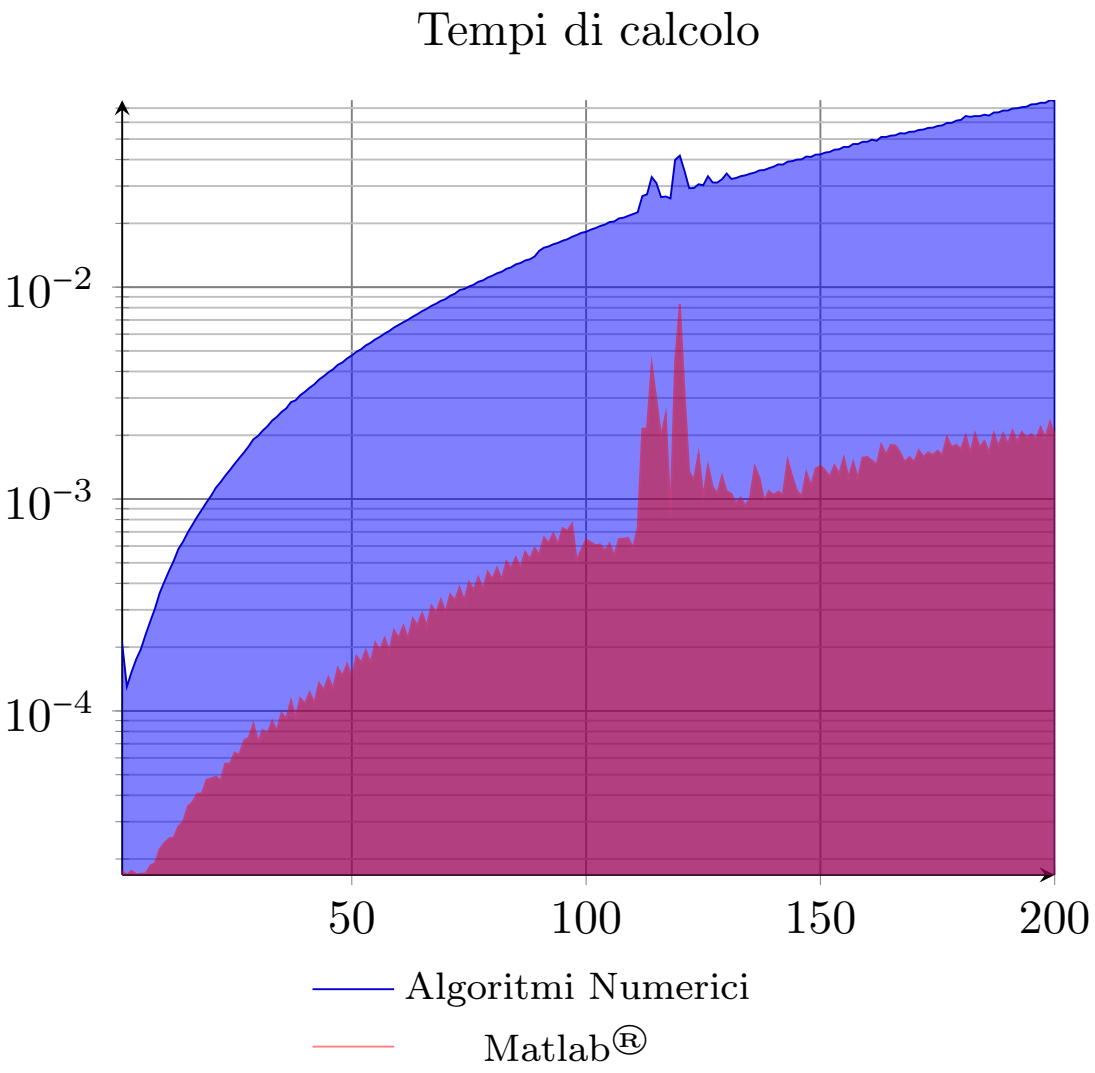


Figura 8.7: Fattorizzazione QR di Gram-Schmidt ($m = n$). Tempi di calcolo.

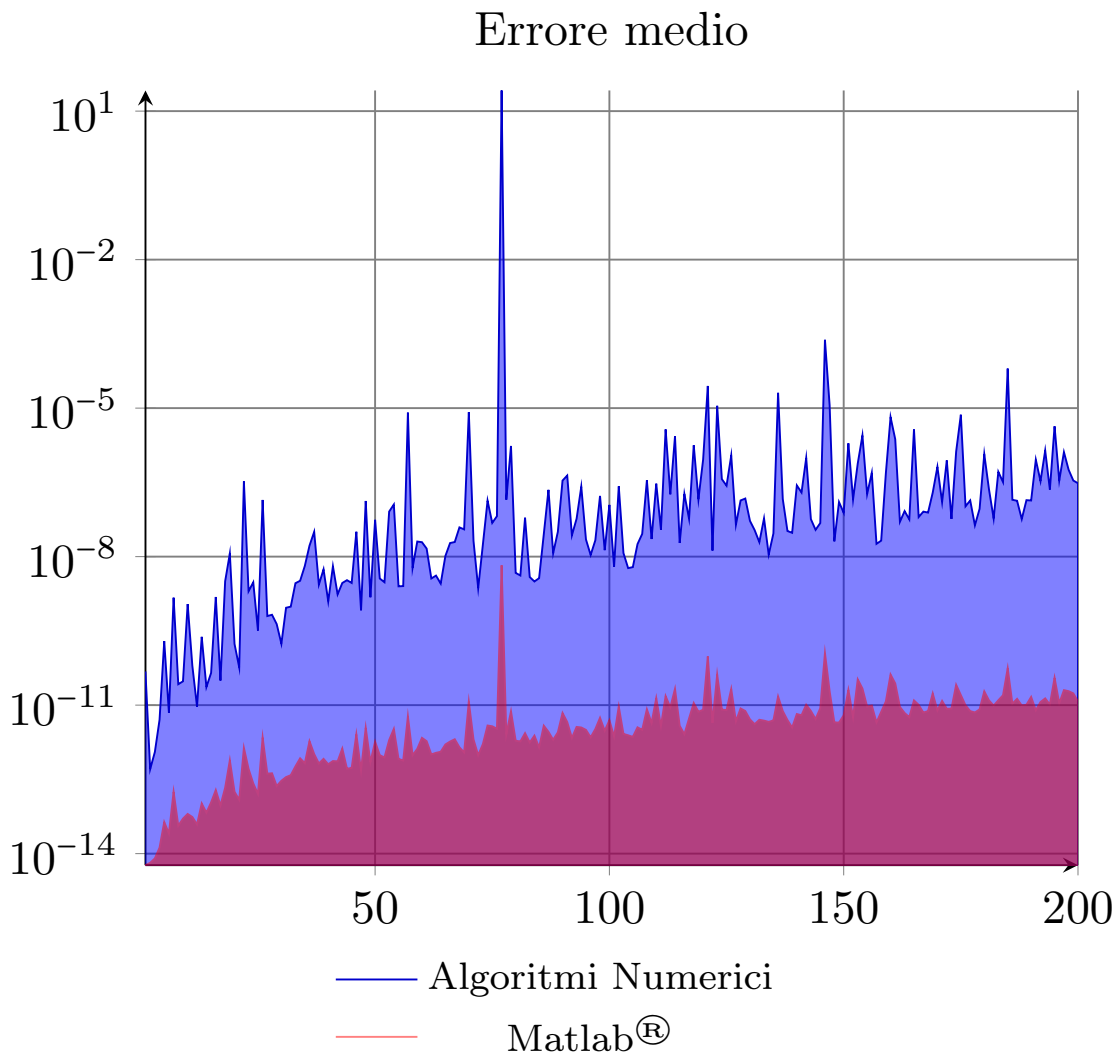


Figura 8.8: Fattorizzazione QR di Gram-Schmidt ($m = n$). Errori di calcolo.

8.3 Problemi minimi quadrati, fattorizzazioni QR

Dopo aver scritto simulato, le singole fattorizzazioni QR, ora le si usa per risolvere un sistema lineare. I grafici mettono a confronto le varie fattorizzazioni, che la legenda riporta.

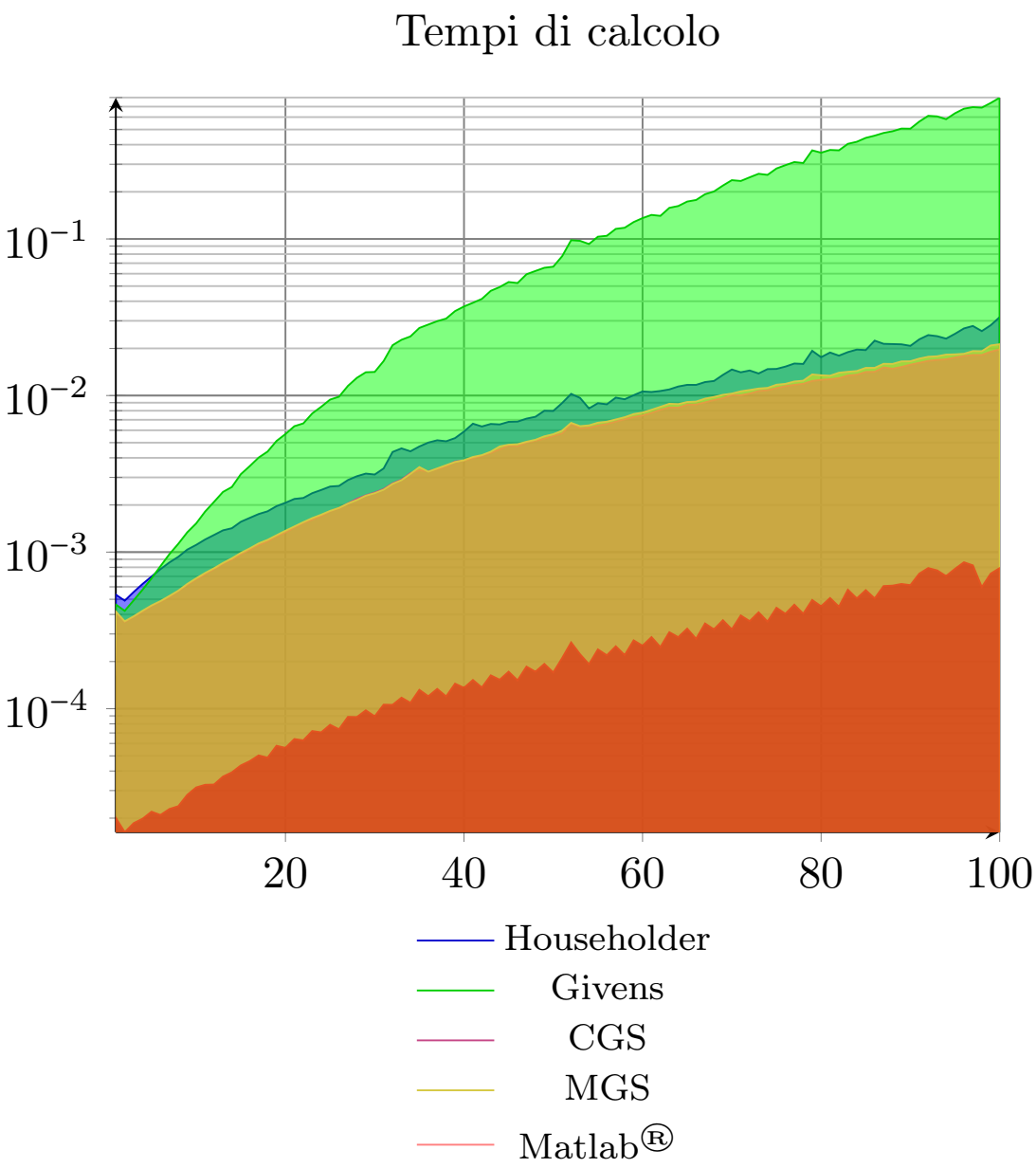


Figura 8.9: Tempo di calcolo della Soluzione Minimi Quadrati $\mathtt{A}(m = n)$.

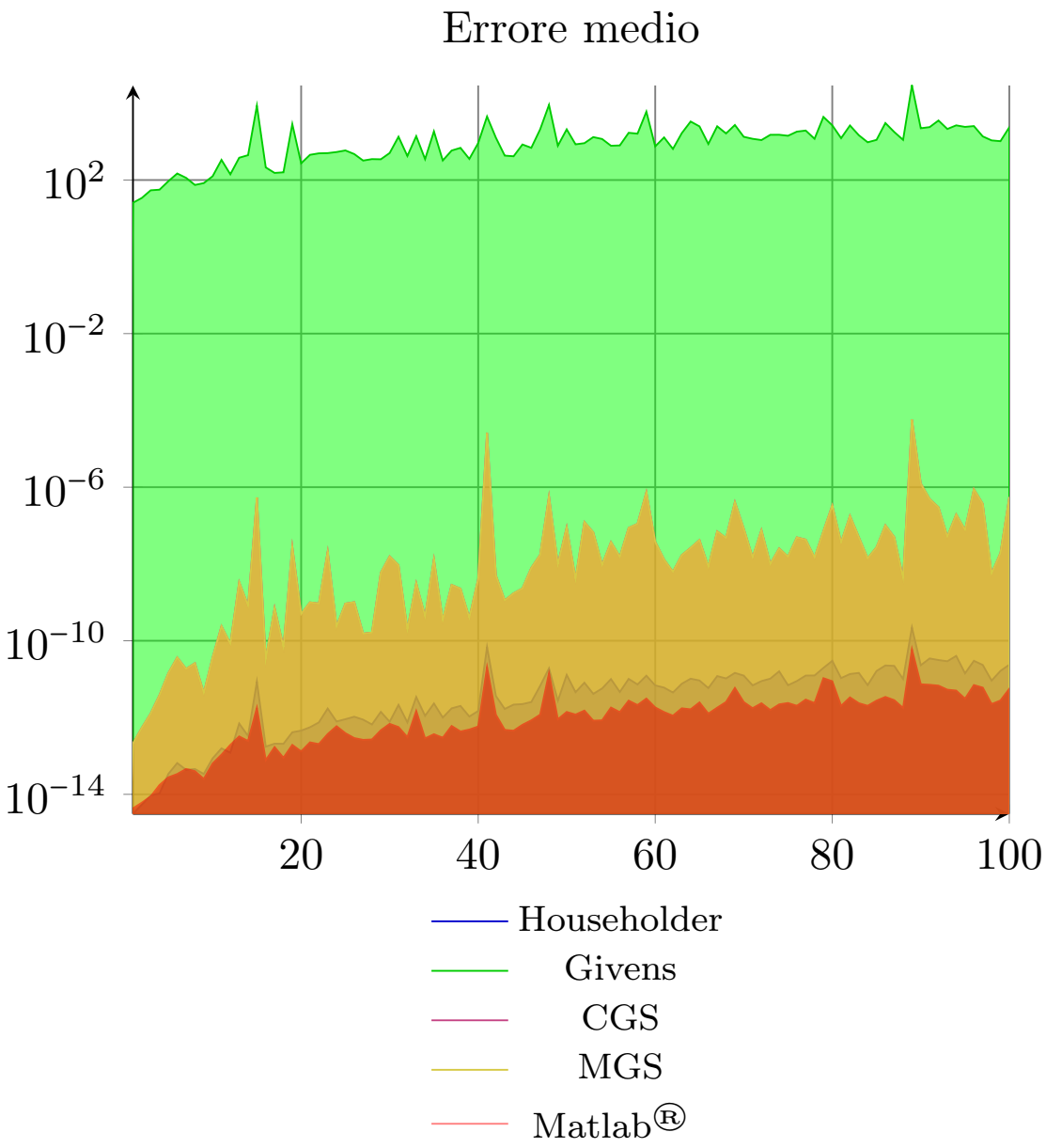


Figura 8.10: Calcolo dell'errore sulla soluzione di un problema ai Minimi Quadrati $\mathtt{A}(m = n)$.

Tabelle riassuntive

Tabella 8.1: Tempo medio di calcolo [s]

Dimensione Matrice	Tempo medio [s]				
	Householder	Givens	CGS	MGS	Matlab
10	0.001112	0.001525	0.000672	0.000680	0.000031
20	0.002066	0.005676	0.001337	0.001369	0.000056
40	0.005890	0.037051	0.003837	0.003863	0.000136
60	0.010618	0.135898	0.007465	0.007783	0.000252
80	0.017573	0.355477	0.012593	0.013468	0.000451
100	0.031643	0.797802	0.019703	0.021267	0.000791

8.4 Metodi iterativi metodo GMRES

Si è voluto congrontare `gmres()` di Matlab® con gli Algoritmi implementati, la matrice è sempre random piena non simmetrica dunque il metodo di Arnoldi, le iterazioni arrivano a n dimensione della matrice. La matrice va da 3 a 100. Anche qui tempi e errori di calcolo, e infine la tabella riassuntiva dei tempi di calcolo.

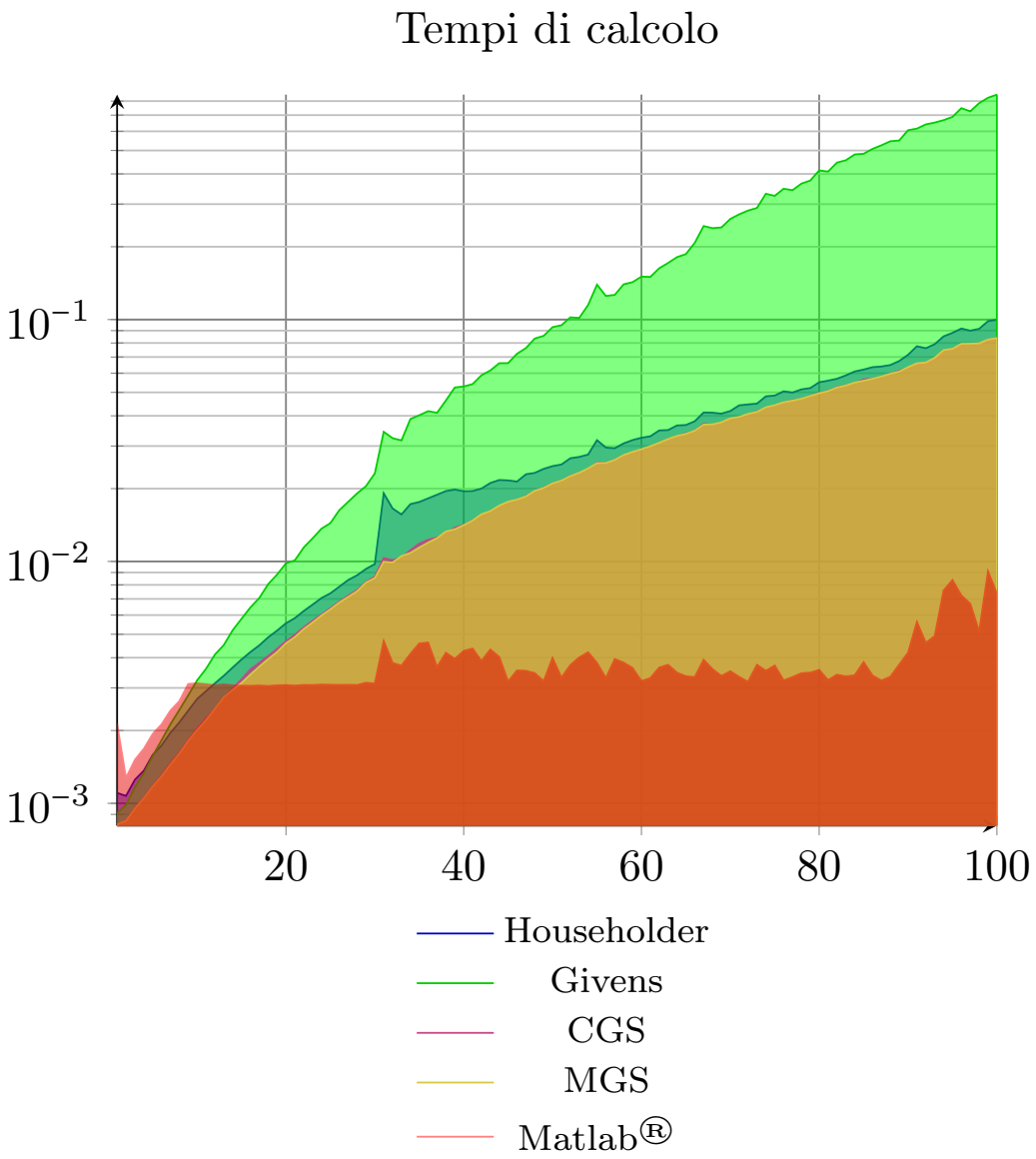


Figura 8.11: Tempo di calcolo della Soluzione Minimi Quadrati `A`($m = n$).

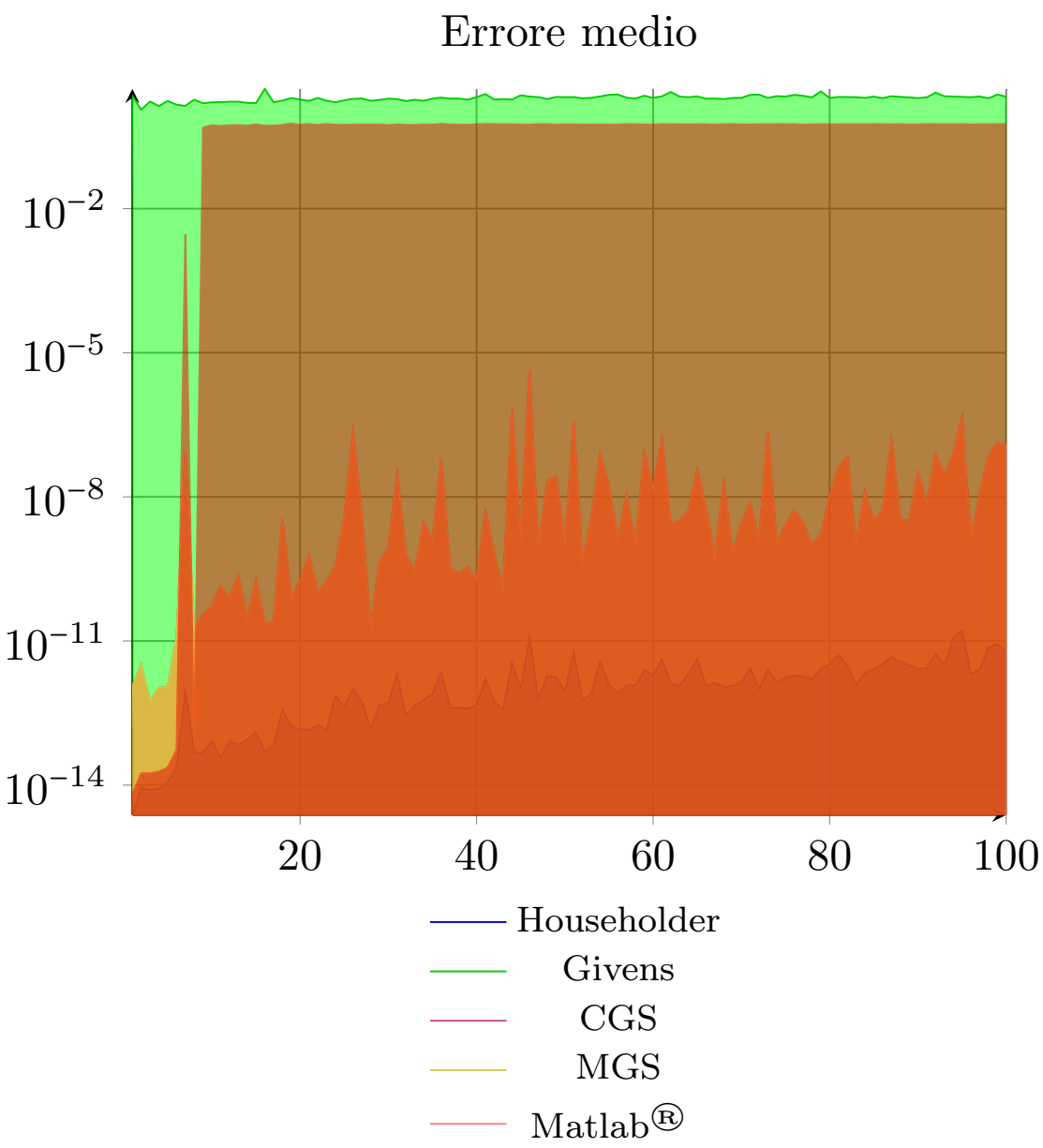


Figura 8.12: Calcolo dell'errore sulla soluzione di un problema ai Minimi Quadrati $\texttt{A}(m = n)$.

Tabelle riassuntive

Tabella 8.2: Tempo medio di calcolo [s]

Dimensione Matrice	Tempo medio [s]				
	Householder	Givens	CGS	MGS	Matlab
10	0.002699	0.003225	0.002036	0.002009	0.003136
20	0.005551	0.009822	0.004680	0.004587	0.003081
40	0.019506	0.052861	0.014151	0.014120	0.004268
60	0.032462	0.150477	0.028941	0.029117	0.003201
80	0.055053	0.413455	0.049643	0.049573	0.003562
100	0.099773	0.850902	0.082725	0.083850	0.007358

Riferimenti bibliografici

[Rod08] Giuseppe Rodriguez. *Algoritmi numerici*. Cagliari: Pitagora Editrice Bologna, 2008.

Siti web consultati

[Wiki15a] *Oskar Perron*. 2015. URL: https://it.wikipedia.org/wiki/Oskar_Perron.

[Wiki15b] *Issai Shur*. 2015. URL: https://it.wikipedia.org/wiki/Issai_Schur.

[Wiki16a] *Walter Edwin Arnoldi*. 2016. URL: https://en.wikipedia.org/wiki/Walter_Edwin_Arnoldi.

[Wiki16b] *André-Louis Cholesky*. 2016. URL: https://it.wikipedia.org/wiki/AndrÃĺ-Louis_Cholesky.

[Wiki16c] *Jacques Solomon Hadamard*. 2016. URL: https://it.wikipedia.org/wiki/Jacques_Hadamard.

[Wiki16d] *Charles Hermite*. 2016. URL: https://it.wikipedia.org/wiki/Charles_Hermite.

[Wiki16e] *John William Strutt*. 2016. URL: https://it.wikipedia.org/wiki/John_William_Strutt_Rayleigh.

[Wiki17a] *Friedrich Ludwig "Fritz" Bauer*. 2017. URL: https://en.wikipedia.org/wiki/Friedrich_L._Bauer.

[Wiki17b] *Augustin-Louis Cauchy*. 2017. URL: https://it.wikipedia.org/wiki/Augustin-Louis_Cauchy.

[Wiki17c] *Ferdinand Georg Frobenius*. 2017. URL: https://it.wikipedia.org/wiki/Ferdinand_Georg_Frobenius.

[Wiki17d] *Johann Friedrich Carl Gauss*. 2017. URL: https://it.wikipedia.org/wiki/Carl_Friedrich_Gauss.

[Wiki17e] *Jørgen Pedersen Gram*. 2017. URL: https://it.wikipedia.org/wiki/J\%C3\%B8rgen_Pedersen_Gram.

[Wiki17f] *David Hilbert*. 2017. URL: https://it.wikipedia.org/wiki/David_Hilbert.

[Wiki17g] *Nikolay Mitrofanovich Krylov*. 2017. URL: https://en.wikipedia.org/wiki/Nikolay_Mitrofanovich_Krylov.

[Wiki17h] *Cornelius (Cornel) Lanczos*. 2017. URL: https://en.wikipedia.org/wiki/Cornelius_Lanczos.

[Wiki17i] *Pierre-Simon Laplace*. 2017. URL: https://it.wikipedia.org/wiki/Pierre_Simon_Laplace.

[Wiki17j] *Lewis Fry Richardson*. 2017. URL: https://en.wikipedia.org/wiki/Lewis_Fry_Richardson.

[Wiki17k] *Erhard Schmidt*. 2017. URL: https://it.wikipedia.org/wiki/Erhard_Schmidt.