Appunti di Metodi Numerici

A.A. 2020-2021

Matteo Brunello

Contents

1 Introduzione all'analisi numerica

L'analisi numerica e' la materia o branca della matematica che si occupa della definizione e l'analisi di algoritmi per la risoluzione di *problemi* matematici in cui sono coinvolte variabili reali o complesse.

La soluzione di tali problemi rappresenta un'approssimazione della soluzione reale del problema.

Lo studio degli errori costituisce una parte importante dell'analisi numerica, poiche' condizionano in modo diretto la qualita' della soluzione. Una tipologia di errore ineliminabile e' quella degli **errori sperimentali**, cioe' tutti quegli errori che dipendono dalle condizioni dell'esperimento in cui sono stati raccolti i dati e dalla precisione finita degli strumenti con cui e' stata effettuata la misurazione. Altre tipologie di errore possono essere gli **errori di troncamento**, **errori di arrotondamento**, e gli **errori di discretizzazione** che sono introdotti quando i dati sono rappresentati con un numero finito di cifre.

1.1 Buona posizione e condizionamento

Tale modello e' costituito da un insieme di formule che descrivono il comportamento del fenomeno sotto studio. Spesso il fenomeno da studiare e' profondamente complesso, per cui anche il suo modello matematico sara' tale. Per questa ragione le equazioni e formule del modello risultano essere troppo complicate per essere risolte con metodi diretti, e in questi casi si preferisce associare al modello matematico un **modello numerico**. Il fine del modello numerico e' quello di ottenere una forma del modello matematico particolarmente adatta per la risoluzione mediante calcolatore. Questa forma si ottiene introducendo delle semplificazioni o approssimazioni nel modello matematico di partenza.

Un determinato problema numerico viene detto **ben posto** quando questo possiede una e una sola soluzione che dipende con *continuita' dai dati*. In caso contrario viene detto **mal posto**.

Esempi di problemi mal posti possono essere:

1. Trovare $x \in \mathbb{R}$ che soddisfa l'equazione $x^2 + 1 = 0$ (*la soluzione non esiste*)
2. Trovare $x, y \in \mathbb{R}$ che soddisfano l'equazione $x + y = 1$ (*manca l'unicita'*)

Ma cosa significa *con continuita' dai dati*? Semplicemente significa che se ho un determinato dato $d' \neq d$ in input a cui corrisponde una determinata soluzione $x' \neq x$, allora se $d' \to d$ risulta che $x' \to x$. Quando questo accade si dice che il *problema numerico* in questione e' *stabile* cioe' che piccole perturbazioni sui dati in ingresso non influenzano in modo significativo la soluzione (dati in uscita).

Nell'analisi numerica spesso si vogliono avere delle stime qualitative sui modelli numerici che indichino quanto la soluzione venga influenzata dai dati perturbati in input.
Questa caratterizzazione viene detta **condizionamento del problema** ed e' definita come:

> ***Definizione**: Sia δd una perturbazione dei dati d di un problema e sia δx la corrispondente perturbazione sulla sua soluzione x. Sia inoltre $\|\cdot\|$ una qualsiasi norma vettoriale. Il **numero di condizionamento***

assoluto $K = K(d)$ *e' definito dalla relazione:*

$$\|\delta x\| \leq K \|\delta d\|$$

mentre per il ***numero di condizionamento relativo*** $k = k(d)$*:*

$$\frac{\|\delta x\|}{\|x\|} \leq k \frac{\|\delta d\|}{\|d\|}$$

Il condizionamento misura quindi **quanto** un errore sui dati possa essere amplificato nei risultati. Piu' tale numero e' grande piu' risulta essere amplificato l'errore nei risultati.

1.2 Algoritmi

> ***Definizione****: Un* ***algoritmo*** *e' una sequenza univoca di un numero finito di operazioni elementari che stabilisce come calcolare la soluzione di un problema assegnati certi dati iniziali.*

Un algoritmo e' detto **stabile** quando al tendere delle operazioni all'infinito, la soluzione tende a quella reale. E' detto **instabile** quando gli errori si propagano in modo incontrollato man mano che l'algoritmo viene eseguito producendo un risultato diverso da quello reale. La complessita' di un algoritmo numerico e' calcolata in flops (*Floating Points Operations Per Second*), cioe' in numero di operazioni in virgola mobile necessarie a risolvere il problema numerico che risolve l'algoritmo. E' utile anche dare una definizione di errore relativo e assoluto per gli algoritmi numerici:

> ***Definizione****: Siano a il valore esatto della soluzione di un problema e* a^* *il valore perturbato di a,* ***l'errore assoluto*** *di un algoritmo e' definito come*
>
> $$\varepsilon = |a - a^*|$$
>
> *mentre* ***l'errore relativo*** *e' definito come*
>
> $$\rho = \frac{|a - a^*|}{|a|}$$

L'errore relativo fornisce informazioni sul numero di cifre significative esatte in a^* (cioe' quelle a partire dalla prima cifra diversa da 0)

2 Numeri di macchina

Come gia' detto in precedenza, gli algoritmi studiati dall'analisi numerica operano sistematicamente su dati affetti da errori. In questa sezione verra' mostrato come vengono memorizzati i numeri reali su un calcolatore e quali sono le principali fonti di errore di tali numeri.

2.1 Origine e causa degli errori

Come gia' detto in precedenza, la presenza di errori e' dovuta a varie cause quali:

- Errata modellizzazione o presenza di errori sui dati sperimentali
- Semplificazioni introdotte dalla conversione a modello matematico in modello numerico
- Memorizzazione dei dati in formato digitale (finito) su calcolatore

Avere quindi un metodo per misurare un errore e' fondamentale nello studio dei problemi numerici. Definiamo quindi due quantita':

> ***Definizione**: Sia α la quantita' da stimare e α^* la quantita' stimata. Definiamo come **errore assoluto**:*
>
> $$\varepsilon = |\alpha - \alpha^*|$$
>
> *e come **errore relativo**:*
>
> $$\rho = \frac{|\alpha - \alpha^*|}{\alpha}$$

In altri termini meno formali, l'*errore relativo* ci da informazioni sul numero di cifre esatte in α^* rispetto ad α.

Nota: E' importante inoltre specificare che se la quantita' da stimare non e' uno scalare, ma un elemento di uno spazio lineare da $\mathbb{R}$, o $\mathbb{C}$, allora al posto del *valore assoluto* si utilizza una *norma vettoriale*.

2.2 Numeri di macchina

Diamo ora una definizione formale dei numeri macchina:

> ***Definizione**: Definiamo come **insieme dei numeri macchina** in base β, con t cifre significative ed esponente nell'intervallo $[L, U]$ l'insieme*
>
> $$\mathbb{F}(\beta, t, L, U) = \{0\} \cup \{x \in \mathbb{R} : x = sign(x) \cdot m \cdot \beta^p\}$$
>
> *dove*
>
> - t e β sono interi positivi, con $\beta \geq 2$
> - $sign(x) = \pm 1$ a seconda del segno di x

- la quantita' m e' la **mantissa**
- p e' un intero compreso tra L e U, detto esponente o caratteristica

I numeri di macchina, spesso vengono memorizzati in una forma chiamata **normalizzata** tramite un processo di *normalizzazione*. Cioe' consiste nell'evitare di memorizzare gli zeri che eventualmente precedono delle cifre significative in modo da sfruttare meglio lo spazio di memorizzazione e rendere la rappresentazione univoca.

Esempio: Sia $\beta = 10$

$$123000 \to 0.123 \cdot 10^6$$
$$0.00000123 \to 0.123 \cdot 10^{-5}$$

L'insieme di *numeri di macchina* $\mathbb{F}$ e' un sottoinsieme proprio di $\mathbb{R}$ poiche' essendo

$$fl : \mathbb{R} \to \mathbb{F}$$
$$x \to fl(x)$$

Che associa ad un qualsiasi numero real il suo corrispondente in $\mathbb{F}$

Dato un qualsiasi $x \in \mathbb{R}$, ci possono essere 3 casi possibili:

- $x \in \mathbb{F}$, il che significa che $fl(x) = x$
- $|x| < \beta^{L-1}$, il che significa che p e' minore del suo limite inferiore L (**underflow**)
- $|x| \geq \beta^U$, pU, si verifica un **overflow**
- $|x| \in [\beta^{L-1}, \beta^U)$ ma $x \notin \mathbb{F}$, cioe' il numero di cifre significative di x e' superiore a t. In questo caso si puo' decidere se **troncare** il valore oppure **arrotondarlo**.

Il **troncamento** e' un'operazione che elimina le cifre in eccesso trascurandole. Il problema di questo approccio e' che gli errori sono tutti positivi, il che, se si effettuasse una somma degli errori, questi si sommerebbero tra loro, mentre se il segno potesse variare questi errori potrebbero cancellarsi con molta probabilita'.

L'**Arrotondamento**, invece, consiste nell'arrotondare alla cifra decimale piu' vicina. Genera un errore assoluto dimezzato rispetto all'operazione di troncamento. Inoltre gode del beneficio che gli errori possono cambiare segno riducendo l'accumulo di errore nella somma.

Arrotondamento unitario *(unit roundoff)*: Quantita' che maggiora l'errore relativo di arrotondamento. Equivale a $u = \frac{1}{2}\beta^{1-t}$

Precisione di macchina *(machine epsilon)* : Quantita' che indica la distanza tra 1 e il successivo numero di macchina. Equivale a: $\varepsilon_M = 2u = \beta^{1-t}$, il che equivale alla seguente definizione:

$$\epsilon_M := min\varepsilon \in \mathbb{R} : trunc(1 + \varepsilon) > 1$$

2.3 Standard IEEE754

Lo standard IEEE754 e' uno standard per la rappresentazione dei numeri in virgola mobile su calcolatore. Lo standard definisce due tipi di precisione:

- **Precisione Singola**: $\mathbb{F}(2, 24, -126, 128)$
- **Doppia Precisione**: $\mathbb{F}(2, 53, -1022, 1024)$

Table 1: Comparazione schematica tra precisione singola e doppia

Precisione	Segno	Esponente	Mantissa	Totali	Range
Singola	1 bit	8 bit	23(+1) bit	32 bit	$10^{-35} - 10^{38}$
Doppia	1 bit	11 bit	52(+1) bit	64 bit	$10^{-324} - 10^{308}$

L'utilizzo delle variabili in singola precisione piuttosto che quelle in doppia precisione e' irrilevante in termini di prestazioni poiche' i calcolatori moderni prima di fare il calcolo trasformano la variabile automaticamente in doppia precisione. L'unico vantaggio e' che possono essere pero' predilette in applicazioni per cui si ha un limitato spazio di memoria.

2.3.1 Propagazione

In luogo delle normali operazioni aritmetiche un calcolatore utilizza le cosiddette operazioni di macchina. In generale, data una operazione di macchina $\oplus$ vale la seguente ipotesi:

$$x \oplus y = fl(x \oplus y) = (x + y)(1 + \varepsilon), |\varepsilon| < u.x, y \in \mathbb{F}$$

Secondo questa ipotesi, l'errore relativo dell'operazione di somma, puo' essere calcolato come:

$$\varepsilon_{x+y} = \frac{|(x + y) - fl(x + y)|}{|x + y|} = \frac{| - \varepsilon_x x - \varepsilon_y y|}{x + y} = \frac{|x|}{|x + y|}|\varepsilon_y| + \frac{|y|}{|x + y|}|\varepsilon_y|$$

Si nota che quando $x + y \to 0$ l'errore cresce tendendo a ∞. La somma puo' diventare quindi **un'operazione pericolosa**. Il numero di condizionamento K dell'operazione somma diventa quindi molto grande quando sommo due numeri all'*incirca uguali in modulo ma con segno discorde* ($x \sim -y$). Quando $x \sim -y$ si dice che si ha una **cancellazione numerica**.

La *moltiplicazione*, invece, non fa crescere in modo incontrollato l'errore poiche' ha coeffieciente $K = 1$

3 Equazioni non lineari

Le equazioni non lineari sono tutte quelle funzioni la cui equazione non e' nella forma $f(x) = ax + b$. Siccome queste funzioni non sono note, per trovarne la soluzione si ricorre ad un algoritmo numerico iterativo per l'approssimazione numerica. (Di fatto anche le calcolatrici ricorrono a tale metodo per il calcolo di alcune funzioni). Formalmente, un problema non lineare si presenta come:

$$f(x) = 0$$

Cioe' trovare il punto x tale che annulli f, o in altri termini significa trovare gli zeri della funzione f. Un possibile approccio per la risoluzione di tali problemi consiste nel generare iterativamente una successione $\{x_1, x_2, ..., x_k, ...\}$ a partire da un punto iniziale x_0, tale che converga ad una radice del problema $\alpha : f(\alpha) = 0$. (gli zeri di una funzione sono anche chiamati *radici*)

Siccome questi metodi iterativi si avvicinano man mano alla soluzione, ogni passo avra' un determinato errore, chiamato **errore al passo** k. Formalmente possiamo definire l'errore di un algoritmo iterativo al passo k, come

$$e_k = x_k - \alpha$$

Avendo quindi definito la nozione di errore, possiamo anche formalizzare il concetto di **convergenza**: Un algoritmo numerico e' convergente quando

$$\lim_{k \to \infty} |e_k| = 0$$

Cioe', se faccio tendere i passi ad infinito, il mio errore sara' sempre piu' piccolo e tendente a zero.

Si puo' inoltre notare che la convergenza puo' essere piu' o meno veloce. Per quantificare la velocita' di convergenza verso la soluzione, introduciamo la seguente

> ***Definizione**: Un metodo iterativo convergente **ha ordine** p se esiste una costante finita C tale che*
>
> $$\lim_{k \to \infty} \frac{|e_{k+1}|}{|e_k|^p} = C$$

Maggiore e' l'ordine p, maggiore e' la riduzione dell'errore che si verifica ad ogni passo, per cui e' minore il numero di iterazioni totali necessarie per raggiungere la precisione richiesta. Il valore del limite C viene detto **costante asintotica dell'errore**. Vale inoltre la pena notare alcune caratteristiche quali:

- Quando un metodo ha ordine $p = 1$ allora deve verificarsi $C \leq 1$.
- Quando $p = 1$ e $C = 0$ si dice che il metodo e' **superlineare**.

Siccome d'ora in avanti useremo largamente i polinomi interpolanti, vale la pena introdurre un teorema fondamentale dell'analisi:

***Teorema (Serie di Taylor)**: Sia $f \in C^{(n+1)[a,b]}$. Allora $\forall x_0 \in [a,b]$ e per ogni intero n esiste un polinomio t_n di grado n (detto **polinomio di Taylor**) tale che:*

$$f(x) = t_n(x) + R_{n+1}(x), x \in [a,b]$$

Il polinomio t_n e' dato da:

$$t_n(x) = \sum_{k=0}^{n} \frac{f^{(k)}(x_0)}{k!}(x - x_0)^k$$

e il resto e' definito come:

$$R_{n+1}(x) = \frac{f^{n+1}(\xi)}{(n+1)!}(x - x_0)^{n+1}$$

per un certo ξ (dipendente da x) apparenente all'intervallo $I_{(x_0,x)}$ di estremi x_0 e x.

Il teorema afferma che data una funzione che sia continua e derivabile fino all'ordine $n+1$, allora esiste per forza un polinomio di grado n che assume gli stessi valori di f (approssima f) nell'intervallo $[a,b]$.

Come ben si sa, inoltre, le funzioni possono avere una o molteplici radici. Questo fatto puo' essere formalizzato dal seguente

***Teorema**: Una radice α ha molteplicita' m se e solo se*

$$f^{(k)}(\alpha) = 0, k = 0, 1, ..., m-1$$
$$f^{(m)}(\alpha) \neq 0$$

Dove $f^{(k)}(\alpha)$ indica la derivata k-esima di f calcolata nel punto α

Una radice di un polinomio e' detta semplice se ha molteplicita' $m = 1$, mentre viene detta multipla se ha molteplicita' $m \geq 2$, o in termini *algebrici*:

*α e' **radice semplice di un'equazione** se si puo' scrivere*

$$f(x) = (x - \alpha)h(x), \text{ con } h(\alpha) \neq 0$$

*α e' invece **radice multipla di un'equazione** con molteplicita' $m \geq 2$ se*

$$f(x) = (x - \alpha)^m h(x), \text{ con } h(\alpha) \neq 0$$

Per studiare le radici di una funzione possiamo anche utilizzare strumenti analitici quali la derivata prima. Studiandone l'andamento, quando essa tende ad essere tangente all'asse delle x ($f'(x) = 0$) significa che si e' in prossimita' di un **minimo** oppure **massimo** relativo della funzione. Il fatto che ci siano piu' massimi e minimi indica la presenza di piu' radici.

Il condizionamento del calcolo delle radici e' dato principalmente dall'andamento della funzione in prossimita' della radice. Difatti, gli algoritmi numerici forniscono un intervallo $[a, b]$ in cui e' contenuta la radice α. E' facile notare che piu' questo intervallo e' grande maggiore sara' l'errore. Se quindi la derivata prima della funzione tende a 0 in prossimita' della radice, l'ampiezza di questo intervallo tendera' a crescere sempre di piu'. Questo perche' il tendere a 0 della derivata prima corrisponde al fatto che la funzione tende ad essere tangente all'asse delle x e quindi ad avere una doppia radice.
Questa conseguenza e' possibile vederla anche in maniera puramente algebrica, applicando il teorema di Taylor si sviluppa $f(x)$ fino al primo termine, ottenendo

$$f(x) = f(\alpha) + f'(\xi)(x - \alpha)$$

se poniamo poi che $|f(x)| < \varepsilon$ si ottiene

$$|x - \alpha| \leq \frac{1}{|f'(\xi)|} \cdot \varepsilon$$

che e' la relazione che esprime il *condizionamento assoluto*. Dalla relazione e' evidente come il valore di f' influenzi direttamente l'andamento dell'errore poiche' compare a denominatore.

3.1 Metodo di bisezione

Il metodo di bisezione e' il metodo numerico piu' semplice e banale dal punto di vista dell'implementazione. L'idea che sta alla base del metodo e' semplicemente quella di andare a dividere progressivamente un intervallo $[a, b]$ che contiene la radice da calcolare. Per dividere l'intervallo se ne calcola il punto medio c. Se il valore della funzione in prossimita' del punto medio e' uguale a 0 la radice e' stata trovata e ci si ferma, in caso contrario si sceglie un intervallo tra $[a, c]$ o $[c, b]$. Per sapere quale dei due intervalli scegliere si determina in quali dei due intervalli e' presente la radice andando a studiare in quali dei due la funzione cambia di segno.

Piu' formalmente, considerata una funzione $f(x)$, continua e contenente un solo zero nell'intervallo $[a, b] : c = \frac{a+b}{2}$ e' il *punto medio* dell'intervallo. Allora si possono verificare le seguenti condizioni:

1. $f(c) = f(\alpha) = 0$
2. $\alpha \in [a, c]$
3. $\alpha \in [c, b]$

Nel primo caso ci si ferma perche' la soluzione e' stata trovata, mentre per determinare se ci si stia trovando nel caso 2 oppure nel caso 3 basta analizzare in quale intervallo la funzione cambia di segno calcolando $f(a) \cdot f(c)$, il cui risultato puo' essere

- Negativo, allora $\alpha \in [a, c]$
- Positivo, allora $\alpha \in [c, b]$

Il metodo prevede poi di reiterare il procedimento ponendo come nuovo intervallo quello scelto (che conterra' sicuramente α).

Per poter calcolare il numero di iterazioni necessarie ad ottenere un **errore inferiore** a ε poniamo

- k: numero dell'iterazione
- x_k: soluzione del passo k, per definizione e' $x_k = \frac{a_k+b_k}{2}$

Allora le iterazioni necessarie sono pari a

$$|x_k - \alpha| \leq \frac{b-a}{2^k} \leq \varepsilon \rightarrow k \geq log_2(b-a) + log_2\varepsilon^{-1}$$

Se supponiamo quindi che la funzione sia continua e che abbia una radice ($f(a) \cdot f(b) < 0$), allora il metodo converge alla soluzione e l'errore si dimezza ad ogni passo poiche' l'ampiezza stessa dell'intervallo viene dimezzata percio' per $k \rightarrow \infty$ si ha che $\varepsilon \rightarrow 0$. Dal momento che l'errore si dimezza ad ogni iterazione k, il metodo ha ordine $p = 1$ e costante asintotica $C = \frac{1}{2}$.

3.2 Metodo di Newton

L'idea che sta alla base del metodo di Newton consiste nel scegliere inizialmente un punto x_0 e di considerarne la sua *retta tangente* rispetto ad f. Questa retta tangente avra' un'intersezione con l'asse x che rappresentera' il nuovo punto da considerare x_1. In questo modo, applicando questi passaggi iterativamente, si puo' ottenere una successione di punti x_k che (sotto oppurtune condizioni) convergono man mano alla radice α. E' possibile formalizzare il generico passo del metodo come:

$$x_{k+1} = x_k - \frac{f(x_k)}{f'(x_k)}$$

Questa formalizzazione si ottiene considerando inizialmente un generico fascio di rette con centro nel punto di coordinate $(x_k, f(x_k))$, e successivamente imponendo che il coefficiente angolare sia uguale a $f'(x_k)$ e che $y = 0$.

> **Nota**: f' deve essere sempre $\neq 0$, ma questo accade per le ipotesi iniziali (cioe' che f sia continua e dotata di una sola radice), poiche' in caso contrario, f corrisponderebbe ad una funzione orizzontale che non ha intersezioni con l'asse delle ascisse.

Si puo' derivare l'errore generico al passso $k+1$ costruendo il metodo tramite lo sviluppo in serie di Taylor e considerandone il resto. L'errore risulta quindi essere

$$|e_{k+1}| \leq \frac{1}{M}|Me_0|^{2k+1}$$

Cio' significa che il metodo converge (e quindi $e \to 0$) se $|Me_0| < 1$, il che avviene quando

$$x_0 \in (\alpha - \frac{1}{M}; \alpha + \frac{1}{M})$$

Cioe' quando il punto iniziale x_0 viene scelto sufficientemente vicino alla soluzione α.

Per quanto riguarda la velocita' di convergenza del metodo si ha che

$$\lim_{k\to\infty} \frac{|e_{k+1}|}{|e_k|^2} = \frac{1}{2}\frac{|f''(\alpha)|}{|f'(\alpha)|}$$

Dalla formula emerge quindi che $p = 2$. Il limite converge ssse α e' una radice semplice (quindi $f(\alpha) \neq 0$). In corrispondenza di radici multiple, la velocita' di convergenza $C = \frac{1}{2}$, quindi $p = 1$ (e quindi estremamente lenta).

Come **criterio di arresto** assegnati una tolleranza τ e un numero massimo di iterazioni N, possiamo utilizzare la seguente formula

$$|x_k - x_{k-1}| < \tau|x_k| \text{ or } f(x_k) = 0 \text{ or } k > N.$$

Il numero massimo di iterazioni viene utilizzato per evitare che il metodo giri all'infinito in situazioni in cui non converga alla soluzione.

Vediamo ora alcuni metodi che si basano sulle idee del metodo di Newton ma che considerano rette con coefficienti angolari differenti. Tali metodi sono appunti definiti metodi *quasi Newton*.

3.3 Metodo delle corde

Alternativamente al metodo di Netwon che considera tutte le rette con tangenti a f (e quindi con coefficiente angolare che dipende da f'), il metodo delle corde considera un valore costante m_k come coefficiente angolare.
Tale metodo, seppur semplice dal punto di vista implementativo, presenta delle prestazioni molto poco soddisfacenti nemmeno se comparate con il metodo di bisezione.

3.4 Metodo delle secanti

Il ragionamento e' analogo a quello del metodo delle corde, ma si utilizza un coefficiente angolare m differente, definito come

$$m_k = \frac{f(x_k) - f(x_{k-1})}{x_k - x_{k-1}}$$

In altri termini, si trova la retta con coefficiente angolare tale che sia secante (passante) per due punti dati. Contrariamente ai metodi Newton e delle corde, questo metodo necessita di due punti iniziali anziche' uno.

In termini algebrici possiamo definire il criterio per la generazione dei punti come

$$x_{k+1} = \frac{x_{k-1}f(x_k) - x_k f(x_{k-1})}{f(x_k) - f(x_{k-1})}$$

Un vantaggio di questo metodo e' che approssima molto bene il metodo delle tangenti (man mano che l'intervallo si rimpicciolisce, i due punti tenderanno a esser molto vicini, rendendo le rette quasi delle tangenti), ma senza la necessita' di dover computare delle derivate prime in modo diretto, le quali potrebbero essere non note a priori. Inoltre, guardando la formula per la generazione della successione dei punti, si nota che e' presente $f(x_{k-1})$, cio' significa che tale valore e' stato calcolato in precedenza e tramite opportune memorizzazioni e' possibile evitare di ricalcolarlo.

E' possibile dimostrare che l'**ordine di convergenza** e' pari a $p = \frac{1+\sqrt{5}}{2} \approx 1.618$ (questa quantita' e' nota come *rapporto aureo*). Cio' indica che a parita' di costo computazionale esso produce una riduzione dell'errore maggiore del metodo di Newton, oltre a non richiedere la computazione della derivata prima.

3.5 Metodi di punto fisso

Fino ad ora i metodi che sono stati trattati (eccetto quello delle secanti) possono essere generalizzati nella forma

$$x_{k+1} = g(x_k)$$

Dove g e' un'opportuna funzione d'iterazione che dipende dal metodo. Se questa funzione converge alla radice si ha quindi che

$$\alpha = g(\alpha)$$

Cioe', la funzione di iterazione come prossimo punto data la radice, ritorna la radice stessa. In analisi, il valore che una funzione mappa in se stesso e' detto **punto fisso**. I metodi di punto fisso quindi non consistono piu' nel trovare un valore α tale che $f(\alpha) = 0$ ma invece che $\alpha = g(\alpha)$. Ovviamente, la funzione g viene scelta in modo che il suo punto fisso coincida con la radice di f.
Il metodo consiste nel porre inizialmente che $x = g(x)$ e di dare un punto iniziale inziale x_0 che sia un'approssimazione di α. Successivamente si procede per costruzione di una successione tramite una regola del tipo

$$x_{k+1} = g(x_k)$$

Si puo' notare che ponendo $x = g(x)$ (in altri termini esplicitando la x di $f(x)$) possiamo ottenere diverse funzioni di iterazione g in base al modo in cui si opera

algebricamente. Non tutte le possibili funzioni di iterazioni pero' garantiscono la convergenza alla soluzione, anzi, piu' precisamente la nozione di convergenza di un metodo iterativo di punto fisso, e' connessa al concetto di **contrattivita'**. La contrattivita' esprime la capacita' di una funzione g di avvicinare tra loro due punti (*contrarre=diminuire le distanze*).

> ***Definizione**: Una funzione $f(x)$ e' contrattiva nell'intervallo $I \subset \mathbb{R}$ se esiste una costante $C \in]0,1[$ tale che*
>
> $$|g(x) - g(y)| \leq C|x - y|, \ \forall x, y \in I$$

Diciamo quindi che il metodo converge *se e solo se* $g(x)$ e' una funzione contrattiva.
Una funzione contrattiva e' anche continua, ma non e' necessariamente derivabile. Nel caso in cui essa sia derivabile risulta che

$$C = \max_{x \in I} |g'(x)|$$

E quindi possiamo dire che la condizione di contrattivita' per una funzione derivabile risulta essere

$$|g'(x)| < 1 \text{ in } I$$

In caso non si verificasse questa condizione, il metodo **puo'** non convergere alla soluzione α. Piu' $|g'(x)| \to 0$ piu' e' veloce la convergenza alla soluzione. Questa relazione tra contrattivita' e convergenza del metodo e' chiarita dal seguente teorema

> ***Teorema**: Sia $f : [a,b] \to [a,b]$ una funzione di classe $C^1[a,b]$ con*
>
> $$|g'(x)| \leq C < 1, \forall x \in [a,b]$$
>
> *e si consideri il metodo iterativo $x_{k+1} = g(x_k)$ con punto iniziale $x_0 \in [a,b]$ Allora:*
>
> - La successione degli x_k converge ad un limite α per $k \to \infty$
> - $\alpha \in [a,b]$
> - α e' l'unico punto fisso di g
> - La convergenza e' almeno lineare e $\frac{x_{k+1}-\alpha}{x_k-\alpha} \to g'(\alpha)$

Il seguente corollario mostra sotto quali condizioni la convergenza del metodo e' quadratica

> ***Corollario:** Sotto le ipotesi del teorema precedente, se $g'(\alpha) = 0$ e $g''(x)$ e' continua in $[a,b]$, allora il metodo iterativo ha ordine almeno 2.*

3.6 Esercizi proposti

Esercizio 1: Applicare il metodo delle tangenti alla funzione:

$$f(x) = \begin{cases} \sqrt{x}, & x \geq 0 \\ -\sqrt{-x}, & x < 0 \end{cases}$$

la cui radice e' $\alpha = 0$.

Soluzione: Calcoliamo in primo luogo la derivata prima di f:

$$f'(x) = \begin{cases} \frac{1}{2\sqrt{x}}, & x > 0 \\ \frac{1}{2\sqrt{-x}}, & x < 0 \end{cases}$$

Costruiamo ora la successione ipotizzando che $x_0 > 0$, ottenendo

$$x_1 = x_0 - \frac{\sqrt{x}}{\frac{1}{2\sqrt{x}}} = x_0 - 2x_0 = -x_0 (x_1 < 0)$$

Proseguiamo calcolando il secondo passo dell'iterazione

$$x_2 = x_1 - \frac{-\sqrt{-x}}{\frac{1}{2\sqrt{-x}}} = x_1 - 2x_1 = -x_1 = x_0 (x_2 > 0)$$

Si nota quindi che la successione generata risulta essere $x_0, -x_0, x_0, \ldots$ e che quindi il metodo non converge.

Esercizio 2: Applicare il metodo delle tangenti alla funzione:

$$f(x) = \begin{cases} \sqrt{x^3}, & x \geq 0 \\ -\sqrt{-x^3}, & x < 0 \end{cases}$$

la cui radice e' $\alpha = 0$.

Soluzione: Si opera analogamente all'esercizio precedente. In primo luogo si calcola la derivata prima di f

$$f(x) = \begin{cases} \sqrt{x^3}, & x \geq 0 \\ -\sqrt{-x^3}, & x < 0 \end{cases}$$

Esercizio 3: Individuare un intervallo che contiene la soluzione positiva dell'equazione

$$e^{-x^2} = x^2$$

Successivamente, stabilire quante iterazioni del metodo di bisezione sono necessarie per determinare tale soluzione a meno di 10^{-3} a partire dall'intervallo precedentemente determinato.

Soluzione: *Da fare*

Esercizio 4: Applicare 3 passi del metodo di bisezione a

$$p(x) = x^2 - cos(x^2) \text{ con intervallo iniziale } [0, 2]$$

Soluzione: *Da fare*

Esercizio 5: Applicare 3 passi del metodo delle secanti a

$$(x-1)^3 = e{-}x^2$$

utilizzando come valori iniziali $x_0 = 0$ e $x_1 = 2$

Soluzione: *Da fare*

Esercizio 6: Il metodo di Newton e' convergente per l'equazione

$$p(x) = x^6 + x^4 + 5x^2 - 12$$

se scegliamo $x_0 = 0$? E con $x_0 = 2$? Calcolare la terza approssimazione della successione di Newton

Soluzione: *Da fare*

4 Approssimazione di funzioni

In alcuni casi l'espressione di una funzione non e' nota a priori o e' in una forma molto complicata e difficile da trattare. Per questa ragione, in questi casi si preferisce darne un'approssimazione mediante un insieme di punti di coordinate $(x_i, y_i), i = 0, ..., n$ tali che $y_i = f(x_i)$. Approssimare una funzione significa quindi trovare la regola generica che descrive una funzione dato un suo sottoinsieme di punti. I due approcci al problema che verranno trattati sono rispettivamente l'**interpolazione** e **l'approssimazione ai minimi quadrati**.

4.1 Interpolazione

L'interpolazione e' un metodo che consiste nell'individuare delle funzioni $f(x)$ passanti per tutti i punti appartententi all'insieme noto. Piu' precisamente, dato l'insieme di punti $(x_i, y_i), i = 0, ..., n$, si dice che una funzione $\phi(x)$ **interpola** i punti se $\phi(x_i) = y_i,\ i = 0, ..., n$.
L'obiettivo e' quello di ottenere la funzione interpolante ϕ come combinazione lineare di $n + 1$ funzioni φ_j con $j = 0, ..., n$.

$$\phi(x) = \sum_{j=0}^{n} a_j \varphi_j(x)$$

Per ottenere i coefficienti a_j della funzione e' sufficiente imporre la condizione di interpolazione $(a_j \varphi_j(x_i) = y_i)$ ottenendo cosi' la formalizzazione del problema:

$$\sum_{j=0}^{n} a_j \varphi_j(x_i) = y_i$$

La cui soluzione e' il vettore dei coefficienti a.

4.1.1 Interpolazione Polinomiale

L'interpolazione polinomiale e' un caso specifico del metodo dell'interpolazione che utilizza come funzioni φ_i dei polinomi $p_n(x)$. La ragione per cui si utilizzano i polinomi e' chiarita formalmente dal seguente teorema:

> ***Teorema (Weierstrass)*** *Sia $f \in C[a, b]$. Per ogni $\varepsilon > 0$ esiste un intero n e un polinomio P_n di grado n tale che*
>
> $$\|f - P_n\|_\infty < \varepsilon$$
>
> (*o alternativamente*)
>
> $$|f(x) - P_n(x)| < \varepsilon$$

Esso afferma che tramite un polinomio qualsiasi e' possibile approssimare *bene quanto si vuole* una funzione continua.

***Teorema**: Il polinomio interpolante esiste ed e' unico se e solo se $x_i \neq x_j$ per $i \neq j$.*

Dimostrazione: Sia $P_n = \sum_{k=0}^{n} a_k x^k$ un polinomio qualsiasi. Applicando la condizione di interpolazione si ottiene che $P_n(x_i) = y_i$ con $i = 0, ..., n$, da cui si ottiene il sistema lineare

$$\sum_{k=0}^{n} a_k x_i^k = y_i \text{ con } i = 0, 1, .., n$$

Questo sistema puo' essere visto in forma matriciale ($Va = y$) come:

$$\begin{pmatrix} 1 & x_0 & x_0^2 & ... & x_0^n \\ 1 & x_1 & x_1^2 & ... & x_1^n \\ ... & ... & ... & ... & ... \\ 1 & x_n & x_n^2 & ... & x_n^n \end{pmatrix} \begin{pmatrix} a_0 \\ a_1 \\ ... \\ a_n \end{pmatrix} = \begin{pmatrix} y_0 \\ y_1 \\ ... \\ y_n \end{pmatrix}$$

Dove la matrice **V** dei termini x_n^k e' detta matrice di *Vandermonde*. Il sistema ammette una e una sola soluzione se e solo se la matrice dei coeffiecienti ha determinante diverso da 0. Tale determinante e' facilmente calcolabile e risulta essere

$$det(V) = \prod_{i,j=0_{i>j}}^{n} (x_i - x_j).$$

Per concludere, si ha che il polinomio esiste ed e' unico se $det(V) \neq 0$, ma questo accade se e solo se i e j sono diversi tra loro, dimostrando il teorema.

Questo teorema risulta utile per stabilire a priori e in modo semplice l'esistenza e unicita' del polinomio interpolante, dal momento che il calcolo del determinante della matrice di Vandermonde e' noto. In secondo luogo, ci permette anche di garantire l'unicita' del polinomio e di poter utilizzare anche altre rappresentazioni che richiederebbero meno risorse computazionali.

L'approccio che consiste nel determinare i polinomi mediante la risoluzione del sistema lineare non e' pero' preferibile per diverse ragioni:

- La matrice V e' malcondizionata se alcuni nodi sono vicini
- Esistono algoritmi con un costo computazionale minore
- La rappresentazione canonica dei polinomi e' instabile, poiche' piccole perturbazioni nei coefficienti a_k possono produrre grandi variazioni sui valori di $p_n(x)$

4.1.2 Interpolazione Lagrangiana

Un approccio possibile per evitare di calcolare direttamente il sistema lineare descritto in precedenza e' quello di utilizzare i cosiddetti *polinomi interpolanti di Lagrange*, definiti come:

$$p_n(x) = \sum_{j=0}^{n} y_j L_j(x)$$

Dove $\{L_j\}_{j=0}^{n}$ sono i *polinomi caratteristici di Lagrange*

$$L_j(x) = \prod_{k=0, k\neq j}^{n} \frac{x - x_k}{x_j - x_k}$$

Tali polinomi caratteristici di Lagrange si annullano nei punti x_i con $i \neq j$, dove j e' l'indice del polinomio caratteristico *j-esimo*. In altre parole, soddisfano la seguente proprieta' di interpolazione:

$$L_j(x_i) = \delta_{ij} = \begin{cases} 1 & i = j \\ 0 & i \neq j \end{cases}$$

In questo modo, il polinomio di Lagrange interpola tutti i punti (x_i, y_i) ed e' quindi una rappresentazione differente dell'unico polinomio interpolante. La valutazione di un punto in un polinomio interpolante di Lagrange ha complessita' temporale di $O(n^2)$, mentre per determinare il polinomio tramite risoluzione del sistema lineare dei coefficienti (utilizzando ad esempio il metodo di Gauss), ha una complessita' di $O(n^3)$.

Per valori di $x \neq x_i$ (poiche' si ha che $f(x_i) = P_n(x_i)$) si puo' valutare l'errore

$$E_n(x) = f(x) - p_n(x)$$

Questo errore E e' anche esprimibile nella forma di resto di Lagrange come:

$$E_n(x) = \frac{f^{n+1}(\xi_x)}{(n+1)!}\omega_n(x)$$

dove $\omega_n(x) = (x - x_0)(x - x_1)...(x - x_n)$ (ω viene chiamato *polinomio nodale*)

Possiamo quindi scrivere che il polinomio P_n approssima f con un certo errore E

$$f(x) = P_n(x) + E_n(x)$$

In cui:

- Se $f(x) \in P_n(x) \rightarrow f^{(n+1)}(x) = 0 \forall x \rightarrow E_n(x) = 0$
- Se $f(x) = 1$ si ha un polinomio di grado 0 per cui $E_n(x) = 0$
- $|E_n(x)| \leq \frac{(b-a)^{n+1}}{(n+1)!} \max_{x\in[a,b]} \left|f^{(n+1)}(x)\right|$

Seguono ora alcune osservazioni che in merito all'errore:

- Il valore assoluto del numeratore $\left|f^{(n+1)}(\xi_x)\right|$ puo' essere maggiorato con una costante. Cio' significa che il rapporto puo' essere reso piccolo a piacere se n e' sufficientemente grande.
- Il secondo fattore $\omega_n(x)$ e' un polinomio di grado $n + 1$ che dipende dalla posizione dei punti (o nodi) $\{x_i\}$, cio' significa che la scelta delle ascisse dei punti di interpolazione influenza l'andamento dell'errore. Se si scelgono nodi equispaziati si ottengono delle oscillazioni sempre piu' grandi al crescere di n.

> **Nota:** L'errore di interpolazione non necessariamente tende a zero quando n tende a infinito.

Cio' implica che il polinomio interpolante non sempre e' una buona approssimazione per una funzione continua. Questo fenomeno e' chiarito dal **teorema di Faber**

> ***Teorema (Faber)**: Per ogni distribuzione dei nodi esiste almeno una funzione $f \in [a,b]$ tale l'errore di interpolazione $\|E_n(f)\|_\infty$ non converga a zero per $n \to \infty$*

Anche se, pero', vale anche il seguente teorema

> ***Teorema**: Per ogni funzione continua esiste almeno una distribuzione dei nodi tale che l'errore di interpolazione converga a zero per $n \to \infty$*

Abbiamo quindi che la distribuzione di nodi influenza direttamente l'errore di interpolazione. La convergenza a zero dell'errore e' assicurata solo se la funzione da interpolare e' derivabile e se i nodi sono scelti secondo una regola particolare. Questo risultato e' garantito dal seguente

> ***Teorema (Bernstein)**: Se $f \in C^1[a,b]$ e se le ascisse di interpolazione $\{x_i\}_{i=0}^n$ sono gli zeri del polinomio di Chebychev di grado $n+1$ alllora l'errore di interpolazione tende a zero per $n \to \infty$*

Un polinomio di Chebychev di grado $n+1$ in $[-1;1]$ puo' essere definito in molteplici modi, ma in questo ambito la definizione piu' conveniente e' la seguente

$$T_{n+1} = cos((n+1)\theta) = 0 \text{ con } x = cos(\theta) \text{ e } \theta \in [0,\pi]$$

Questa rappresentazione del polinomio rende particolarmente agevole calcolarne i suoi zeri, infatti se imponiamo che $T_{n+1} = 0$ si ottiene:

$$T_{n+1}(x) = 0 \to cos((n+1)\theta) = 0 \to x_k = cos(\frac{2k+1}{2n+2}\pi),\ k = \{0,1,..n\}$$

oppure tramite la formula di ricorrenza

$$T_{n+1}(x) = 2xT_n(x) - T_{n-1}(x)$$

Le cui soluzioni sono i cosiddetti **nodi di Chebychev**. E' necessario pero' espandere l'intervallo da $[-1,1]$ in cui e' definito il polinomio di Chebychev ad un generico intervallo in cui e' definito il polinomio interpolante $[a,b]$. Per far cio' e' sufficiente appplicare la seguente trasformazione

$$t_k = \frac{b-a}{2}x_k + \frac{a+b}{2}.$$

4.2 Le costanti di Lebesgue

Il teorema di Weiestrass ci dice che una funzione continua in un intervallo puo' essere approssimata bene quanto si vuole mediante un polinomio di grado sufficientemente elevato. Possiamo dimostrare che esiste un polinomio p_n^* detto **migliore approssimazione uniforme** che minimizza l'errore. Tale polinomio ha l'ulteriore caratteristica di interpolare la funzione su $n+1$ punti distinti le cui ascisse non sono note a priori.
Come abbiamo visto in precedenza, la distribuzione di nodi determina direttamente la convergenza dell'errore di interpolazione. Spesso tale distribuzione viene definita mediante una **matrice di interpolazione**, definita come la matrice triangolare inferiore infinita

$$X = \begin{bmatrix} x_0^{(0)} & & & \\ x_0^{(1)} & x_1^{(1)} & & \\ x_0^{(2)} & x_1^{(2)} & x_2^{(2)} & \\ \vdots & \vdots & \vdots & \ddots \end{bmatrix}$$

In cui la n-esima riga contiene gli $n+1$ nodi da utilizzare per costruire il polinomio interpolante di grado n. Per capire meglio in che modo la matrice di interpolazione influenza l'errore di interpolazione diamo la seguente definizione (e teorema)

> ***Definizione**: Sia P_n l'operatore lineare che a ciascuna funzione f associa il polinomio $P_n f$ che la interpola sui nodi assegnati e sia*
>
> $$E_n(f, X) = ||f - P_n f|\,|_\infty$$
>
> *il corrispondente errore di interpolazione.*
>
> ***Teorema**: Sia $f \in C[a,b]$ e sia X una matrice di interpolazione. Allora*
>
> $$E_n(f, X) \leq (1 + \Lambda_n(X))E_n^*(f),$$
>
> *dove*
>
> $$\Lambda_n(X) = ||\lambda_n(x)|\,|_\infty$$
>
> *sono le **costanti di Lebesgue** e*
>
> $$\lambda_n(x) = \sum_{j=0}^{n} \left| L_n^{(n)}(x) \right|$$
>
> *sono le **funzioni di Lebesgue** e $L_j^{(n)}(x)$* sono i polinomi caratteristici di Lagrange di grado n costruiti sui nodi assegnati.

Questo teorema ci da quindi una stima di quanto e' buona l'approssimazione di un polinomio interpolante (di grado n fissato) comparando tale polinomio con il polinomio di migliore approssimazione.

E' stato dimostrato che con la scelta della matrice di interpolazione T costruita con i nodi di Chebichev si ha che $\Lambda_n(T) < \frac{2}{\pi} log n$, cio' significa che la crescita e' quasi ottimale, cioe' che le costanti di Lebesgue associate alla matrice hanno crescita al piu' logaritmica.
Al contrario, le costanti di Lebesgue per matrici costruite con nodi equispaziati crescono in modo esponenziale, e rappresentano quindi la peggiore scelta di nodi per l'interpolazione.

Le **costanti di Lebesgue** sono importanti inoltre nello studio della stabilita' della valutazione del polinomio di interpolazione poiche' rappresentano il coefficiente di amplificazione dell'errore numerico di interpolazione. Cio' significa che, per matrici di interpolazione per le quali le costanti di Lebesgue hanno un andamento fortemente crescente, l'errore totale del polinomio di Lagrange diverge.

$$\|p_n - \tilde{p}_n\|_\infty \leq \Lambda_n(X) \cdot \max_j |\varepsilon_j|$$

In conclusione, Λ_n gioca il ruolo del numero di condizionamento per il problema dell'interpolazione.

4.3 Polinomio interpolante di Newton

Supponiamo di disporre gia' del polinomio interpolante $p_{n-1}(x)$ che interpola i punti di ascissa $\{x_0, \ldots, x_{n-1}\}$ e di voler aggiungere ai precedenti un punto di interpolazione (x_i, y_i). Per sfruttare il polinomio di cui si dispone, vogliamo cercare di esprimere il polinomio $p_n(x)$ che interpola tutti i punti piu' il punto che si vuole aggiungere

$$p_n(x) = p_{n-1}(x) + g_n(x)$$

per una opportuna funzione $g_n(x)$. Tale funzione (g) deve essere necessariamente un polinomio di grado n. Inoltre vogliamo che interpoli gli stessi punti interpolati da p_{n-1}. In altri termini, questa funzione avra' forma

$$g_n(x) = a_n \cdot \omega_{n-1}(x) = a_n \prod_{i=0}^{n-1} (x - x_i)$$

per determinare il coefficiente a_n imponiamo che p_n interpoli anche l'ultimo punto (x_n, y_n), ottenendo

$$a_n = \frac{y_n - p_{n-1}}{\omega_{n-1}(x_n)}$$

Verrebbe da pensare quindi che sia possibile costruire un polinomio interpolante p_n qualsiasi utilizzando tale procedimento ponendo come polinomio di partenza $p_0(x) = y_0$. Esiste pero' un algoritmo piu' stabile e con minore complessita'.
Osserviamo che a_n e' il coefficiente del polinomio con il grado massimo. Possiamo rappresentare il coefficiente di grado massimo di un polinomio che interpola la funzione $f(x)$ e i punti di ascisse $x_0, \ldots, x_n$ col simbolo

$$a_n = f[x_0, \ldots, x_n]$$

Tale simbolo denota usualmente le cosiddette **differenze divise** di ordine $n+1$ che si calcolano mediante la seguente relazione

$$f[x_0, \ldots, x_n] = \frac{f[x_1, \ldots, x_n] - f[x_0, \ldots, x_n]}{x_n - x_0}$$

Tramite le differenze divise di puo' quindi ottenere il polinomio interpolante di Newton

$$f(x) = P_n(x) + E_n(x)$$

con

$$\begin{aligned} P_n(x) &= f(x_0) + (x - x_0) f[x_0, x_1] + \ldots \\ &\quad + (x - x_0) \ldots (x - x_{n-1}) f[x_0, x_1, \ldots, x_n] \\ E_n(x) &= (x - x_0)(x - x_1) \ldots (x - x_n) f[x, x_0, x_1, \ldots, x_n] \end{aligned}$$

Si e' ottenuta quindi una rappresentazione alternativa del polinomio di Lagrange ma con costo computazionale minore poiche' richiede solamente $\frac{n}{2}$ divisioni e n^2 sottrazioni. Inoltre se si volesse aggiungere un punto, il metodo non richiederebbe di calcolare tutti i polinomi caratteristici come nel metodo di Lagrange.

4.4 Approssimazione ai minimi quadrati

Spesso quando si vuole approssimare una funzione non si hanno dei valori necessariamente distinti, inoltre i dati racconti possono avere dell'errore introdotto. In casi di questo tipo non e' conveniente ricorrere all'interpolazione, dal momento che e' nota una forte presenza di errori sui dati.
L'idea e' quella di trovare per tale insieme di punti, una funzione che sia un'approssimazione del fenomeno rappresentato dai punti $\{(x_i, y_i)\} i = 0, ..., m$ dove m e' il numero di punti. Questa funzione come detto in precedenza puo' essere rappresentata come una combinazione lineare di funzioni $\phi_k(x), k = 0, ..., m$
Per esempio, e' stato visto che nel caso polinomiale $\phi_k(x) = x^k$ l'approssimazione risultante e' un polinomio di grado n: $p_n(x) = c_0 + c_1 x + c_2 x^2 + ... + c_n x^n$
Si vuole pero' considerare il caso piu' generale ed usare quindi $\{\phi_k(x)\}_{k=0,..,n}$ per trovare quindi una funzione della forma:

$$f_n(x) = c_0\varphi_0(x) + c_1\varphi_1(x) + ... + c_n\varphi_n(x) = \sum_{k=0}^{n} c_k\varphi_k(x)$$

con $f_n(x)$ individuato con il criterio dei minimi quadrati, cioe' tutti i c_k tali che:

$$\varepsilon_2 = \sum_{i=0}^{m} [y_i - \sum_{k=0}^{n} c_k\phi_k(x_i)]^2 = \sum_{i=0}^{m} [y_i - f_n(x_i)]^2$$

Per scegliere a quale tipo di famiglia di funzioni appartiene $\phi_k(x)$ si deve vedere se il fenomeno individuato e' un modello esponenziale, periodico o altro. Ci

limiteremo a considerare il caso polinomiale $\phi_k(x) = x^k$, piu' formalmente, si vuole trovare la combinazione lineare $\sum_{k=0}^{n} c_k x_i^k$ tale che

$$\varepsilon_2 = \sum_{i=0}^{m} [y_i - \sum_{k=0}^{n} c_k x_i^k]^2$$

sia minimo. Siccome ε_2 varia al variare dei coefficienti c_k, possiamo considerarla come in funzione di tali coefficienti, quindi si puo' dire che $\varepsilon_2(c_0, c_1, ..., c_n)$. Per trovare il punto di minimo si puo' pensare di derivare ε_2 e porla $= 0$ ottenendo quindi il seguente sistema lineare:

$$\sum_{i=0}^{m} \sum_{j=0}^{n} c_j x_i^{j+k} = \sum_{i=0}^{m} y_i x_i^k \text{ con } k = 0, ..., n$$

Ne risulta quindi un sistema di $(n+1)$ equazioni in $(n+1)$ incognite:

$$\begin{pmatrix} \sum_{i=0}^{m} 1 & \sum_{i=0}^{m} x_i & \sum_{i=0}^{m} x_i^2 & ... & \sum_{i=0}^{m} x_i^n \\ \sum_{i=0}^{m} x_i & \sum_{i=0}^{m} x_i^2 & \sum_{i=0}^{m} x_i^3 & ... & \sum_{i=0}^{m} x_i^{n+1} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ \sum_{i=0}^{m} x_i^n & \sum_{i=0}^{m} x_i^{n+1} & \sum_{i=0}^{m} x_i^3 & ... & \sum_{i=0}^{m} x_i^{2n} \end{pmatrix} \begin{pmatrix} c_0 \\ c_1 \\ \vdots \\ c_n \end{pmatrix} = \begin{pmatrix} \sum_{i=0}^{m} y_i \\ \sum_{i=0}^{m} y_i x_i \\ \vdots \\ \sum_{i=0}^{m} y_i x_i^n \end{pmatrix}$$

Quando $n = 1$ si dice che il sistema e' una retta ai minimi quadrati oppure retta di regressione.

Esempio: Calcolare la retta ai minimi quadrati relativa a

x_i	-1	0	1	2	3	4	5
y_i	10	9	7	5	4	3	0

Abbiamo che $m = 7$, per cui bisogna risolvere il sistema lineare

$$\begin{pmatrix} \sum_{i=0}^{m} 1 & \sum_{i=0}^{m} x_i \\ \sum_{i=0}^{m} x_i & \sum_{i=0}^{m} x_i^2 \end{pmatrix} \begin{pmatrix} c_0 \\ c_1 \end{pmatrix} = \begin{pmatrix} \sum_{i=0}^{m} y_i \\ \sum_{i=0}^{m} y_i x_i \end{pmatrix}$$

Il che, con i dati forniti dal problema equivale a risolvere:

$$\begin{pmatrix} 7 & 20 \\ 20 & 92 \end{pmatrix} \begin{pmatrix} c_0 \\ c_1 \end{pmatrix} = \begin{pmatrix} 37 \\ 25 \end{pmatrix}$$

Il che risulta che $c_0 = 8.6429$ e $c_1 = -1.6071$. In conclusione la retta di regressione risultate sara'

$$p_1(x) = c_0 + c_1 x \to -1.6071x + 8.6429$$

4.5 Esercizi

TODO: Presenti sul file **Esercizi.pdf** nella quarta settimana sulla pagina moodle del corso. Altri esercizi sono anche sul file della lezione in pdf sempre. (Hint: Per vedere come svolgerli guardare la lezione di teoria della stessa settimana)

5 Quadratura Numerica

Un problema di quadratura consiste nel trovare una formula di quadratura che approssimi un integrale definito del tipo

$$I(f) = \int_a^b f(x)dx$$

e' possibile rappresentare tale formula come combinazione lineare dei valori assunti dalla funzione integranda in $n+1$ punti

$$I_n(f) = \sum_{j=0}^{n} \alpha_j f(x_j),$$

In cui:

- I coefficienti a_j sono detti **pesi**
- I punti x_j sono detti **nodi**

Ogni formula di quadratura ha inoltre un determinato grado di precisione, detta *precisione algebrica*. Si dice che una formula di quadratura $I_n(f)$ ha grado di precisione *almeno* r se e' esatta per un polinomio $p \in \Pi_r$. Cioe' se vale la relazione (con p polinomio di grado r):

$$I_n(p) = I(p)$$

Abbiamo ora tutti gli ingredienti necessari per introdurre due nozioni, che rappresentano di fatto i metodi principali con cui costruire una formula di quadratura. Il primo, detto *metodo dei coefficienti indeterminati*, consiste nel fissare arbitrariamente i nodi e determinare i pesi in modo che la formula abbia precisione algebrica n. Quindi, se si usa la base canonica per rappresentare un polinomio di grano n, deve valere la seguente relazione

$$I_n(x^i) = I(x^i) \qquad i = 0, \ldots, n$$

Espandendo le definizioni si ottiene

$$\sum_{j=0}^{n} x_j^i \alpha_j = \frac{b^{i+1} - a^{i+1}}{i+1}, \qquad i = 0, \ldots, n$$

E quindi e' un sistema lineare del tipo $X^T\alpha = b$, dove X e' la matrice di Vandermonde. E' gia' stato visto che tale matrice e' non singolare solo quando i nodi sono tutti diversi tra loro, per cui sappiamo che per ogni precisione algebrica e' possibile costruire una formula di quadratura, posto che le condizioni di non singolarita' siano soddisfatte.

Il secondo metodo per costruire le formule di quadratura, sfrutta in modo diretto il teorema di Weiestrass e quindi i polinomi interpolanti. L'idea e'

quella di utilizzare al posto della funzione integranda un polinomio $P_n(x)$ che la interpola nei nodi scelti. Per cui, sostituendo il polinomio al posto della funzione nell'integrale si ottiene

$$I(f) = \int_a^b \sum_{j=0}^{n} f(x_j) L_j(x) dx$$

Dove per rappresentare p_n e' stata scelta la forma del polinomio interpolante di Lagrange. Per la linearita' dell'integrale, si ottiene quindi che

$$I_n(x) = \sum_{j=0}^{n} f(x_j) \int_a^b L_j(x) dx,$$

Abbiamo quindi trovato che il singolo peso della formula α_j equivale a

$$a_j = \int_a^b L_j(x) dx$$

In questo caso se i nodi sono pari a n la formula avra' precisione algebrica *almeno* n. Inoltre osserviamo che se $f(x) \in \mathbb{P}_n \rightarrow f(x) = \sum_{i=0}^{n} f(x_i) L_i(x)$ con $E_n(f) = 0$ quindi anche l'integrazione e' esatta.

5.1 Formule di Newton-Cotes

Ora che abbiamo un modo per trovare i pesi e poter cosi' costruire una formula interpolatoria su un numero arbitrario di nodi, l'unica incognita rimane proprio la distribuzione dei nodi. Le formule di Newton-Cotes sono formule di quadratura interpolatorie costruite considerando una distribuzione di *nodi equispaziati*.

$$x_j = x_0 + jh, \qquad j = 0, 1, ..., n.$$

Possono essere di due tipi:

- Formule chiuse: L'intervallo di nodi comprende a e b, e quindi $h = \frac{b-a}{n}$
- Formule aperte: L'intervallo di nodi non comprende a e b, percio' $h = \frac{b-a}{n+2}$

Queste formule dipendono solamente da n oltre che dal passo h, e lo si vede impostando un cambio di variabile ($x = x_0 + ht$) nella formula del generico peso α_j, ottenendo nel caso delle formule chiuse

$$\alpha_j = \int_a^b L_j(x) dx = h \int_0^n L_j(x_0 + ht) dt$$

mentre nel caso delle formule aperte

$$\alpha_j = \int_a^b L_j(x) dx = h \int_{-1}^{n+1} L_j(x_0 + ht) dt$$

Per le proprieta' di linearita' dell'integrale, la generica formula di quadratura diventa quindi (raccolgo solo h):

$$I_n(f) = h\sum_{j=0}^{n} f(x_j)\alpha_j, \qquad \text{con } \alpha_j = \int_l^u L_j(x_0 + ht)dt$$

dove

- $l = 0$, $u = n$ in caso di formule chiuse
- $l = -1$, $u = n + 1$ in caso di formule aperte

Tramite l'ultima formula generica, si possono poi derivare le cosiddette formule di Newton-Cotes elementari:

- **Formula dei trapezi (n=1, chiusa)**:

$$\int_a^b f(x)dx = \frac{b-a}{2}[f(a) + f(b)]$$

- **Formula di Simpson (n=2, chiusa)**

$$\int_a^b f(x)dx = \frac{h}{3}[f(a) + 4f(\frac{a+b}{2}) + f(b)], \text{ con } h = \frac{b-a}{2}$$

- **Formula del rettangolo (n=0, aperta)**

$$\int_a^b f(x)dx = b - af(\frac{a+b}{2})$$

In generale, le formule elementari sono utili solo in casi in cui la funzione integranda e' molto semplice e su intervalli molto piccoli. Per ottenere una precisione maggiore nel calcolo dell'integrale spesso si ricorre alle cosiddette *formule composte.*

5.2 Formule composte

Le formule composte sfruttano la proprieta' di linearita' dell'integrale, cioe' che un integrale definito a a b possa essere rappresentato come la somma di tutti i sotto integrali che lo compongono

$$\int_a^b f(x)dx = \sum_{i=1}^{n} \int_{x_{i-1}}^{x_i} f(x)dx$$

L'idea e' quindi quella di "spezzare" l'intervallo per cui bisogna valutare l'integrale in un numero arbitratio di singoli integrali la cui ampiezza dell'intervallo e' h. Tali integrali piu' semplici possono poi essere risolti utilizzando una formula semplice come una di quelle derivate in precedenza di Newton-Cotes.

Il ragionamento puo' essere applicato alle formule di quadratura esposte in precedenza, ottenendo:

- **Formula composta dei trapezi**:

$$I_1^{(c)}(f) = \frac{h}{2}[f(a) + 2\sum_{i=1}^{n-1} f(x_i) + f(b)]$$

- **Formula composta di Simpson**:

$$I_2^{(c)}(f) = \frac{h}{2}[f(a) + 2\sum_{i=1}^{n-1} f(a+2ih) + 4\sum_{i=1}^{n} f(a+(2i-1)h) + f(b)]$$

- **Formula composta del Rettangolo**

$$I_0^{(c)} = h\sum_{i=0}^{n-1} f(\frac{x_i + x_{i+1}}{2})$$

Le formule composte di Newton-Cotes hanno inoltre una particolare caratteristica: oltre che ad avere una precisione algebrica di almeno n (in quanto interpolatorie), ma in caso n sia pari, allora la precisione algebrica e' di almeno $n+1$. Per cui mentre la formula dei trapezi ($n = 1$) ha precisione algebrica 1, le formule del punto medio ($n = 0$) e di Simpson ($n = 2$) hanno precisione algebrica pari a 1 e 3 rispettivamente.

5.3 Errori delle formule

Per dare una stima dell'errore di quadratura nelle formule di quadratura interpolatoria si puo' pensare di rappresentare l'errore in termini di errore interpolatorio. Sia quindi l'errore di interpolazione pari a

$$E_n(x) = \frac{f^{(n+1)}(\xi_x)}{(n+1)!}\omega_n(x)$$

Allora, possiamo rappresentare l'errore di una formula di quadratura interpolatoria come

$$E_n(f) = \int_a^b \frac{f^{(n+1)}(\xi_x)}{(n+1)!}\omega_n(x)$$

Applicando il teorema della media si ottengono le seguenti formule per la stima dell'errore

- **Formula dei Trapezi**
  - Elementare: $-f''(\eta)\frac{(b-a)^3}{12}$
  - Composta: $-f''(\eta)\frac{b-a}{12}h^2$
- **Formula di Simpson**
  - Elementare: $-f^{(4)}(\eta)\frac{[(b-a)/2]^5}{90}$
  - Composta: $-f^{(4)}(\eta)\frac{(b-a)}{180}h^4$
- **Formula del Rettangolo**
  - Elementare: $f''(\eta)\frac{(b-a)^3}{24}$
  - Composta: $f''(\eta)\frac{b-a}{24}h^2$

5.4 Esercizi

5.4.1 Esercizio 1

Si dimostri che la seguente formula di quadratura

$$\int_{-1}^{1} f(x)dx \approx \frac{1}{9}[5f(\sqrt{\frac{3}{5}}) + 8f(0) + 5f(-\sqrt{\frac{3}{5}})]$$

- Ha grado di precisione $r = 5$
- Si applichi tale formula per valutare

$$I = \int_{0}^{1} \frac{sinx}{1+x}dx$$

Soluzione: Per verificare che abbia grado $r = 5$ basta verificare che ponendo $f(x) = x^5$ si ottiene $\int_{-1}^{1} f(x)dx = \frac{1}{9}[5f(\sqrt{\frac{3}{5}}) + 8f(0) + 5f(-\sqrt{\frac{3}{5}})]$.
Mentre per $f(x) = x^6$ l'uguaglianza non e' verificata.

Per il punto 2 bisogna cambiare l'intervallo di integrazione da $[0, 1]$ a $[1, -1]$

5.4.2 Esercizio 2

Determinare i pesi della formula di quadratura

$$\int_{0}^{1} f(x)dx = \alpha_0 f(0) + \alpha_1 f'(0) + \alpha_2 f(1)$$

e calcolare il grado di precisione.

Soluzione: Ci sono 3 α, quindi sappiamo che deve essere esatta per $f(x) = 1, x, x^2$ Hint: Poni a sistema le varie sostituzioni nell'uguaglianza ed esplicita gli α

5.4.3 Esercizio 3

Calcolare con la formula di Simpson semplice e con quella composta su due intervalli l'integrale

$$I = \int_{1}^{2} \sqrt{x}dx$$

e valutare gli errori.

Soluzione: Applico la formula semplice e ottengo:

$$I_1 = \frac{1}{6}[f(1) + f(\frac{3}{2}) + f(2)] \approx 1.2188655$$

Applico la formula composta su 2 intervalli e ottengo: $$

$$

6 Ripasso di nozioni di algebra lineare

In questo capitolo si vuole brevemente ripassare ed eventualmente introdurre concetti fondamentali dell'algebra lineare, fondamentali per capire i prossimi capitoli.
Nel sottocapitolo dei vettori ci limiteremo a dare qualche definizione senza soffermarci troppo sul significato geometrico delle stesse.

6.1 Vettori

Definizione: Una ***combinazione lineare*** dei vettori $\{x_1, \ldots, x_k\}$ con coefficienti $a_i \in \mathbb{R}$, $i = 1, \ldots, k$ e' il vettore $x \in V$ dato da

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

Definizione: $span(x_1, \ldots, x_k)$ e' il sottospazio generato da tutte le possibili combinazioni lineari dei suoi vettori

Definizione: Si dice che i vettori $\{x_1, \ldots, x_k\}$ sono linearmente indipendenti se e nessun vettore e' esprimibile mediante combinazione lineare di altri. In altri termini se vale che

$$\sum_{i=1}^{k} \alpha_i x_i = 0 \ \rightarrow \ \alpha_i = 0, i = 1, \ldots, k$$

Definizione: Una ***base*** e' un insieme di vettori linearmente indipendenti che genera l'intero spazio vettoriale. La cardinalita' di una base ne determina anche la ***dimensione*** dello spazio vettoriale.

Definizione: Uno **spazio normato** e' uno spazio vettoriale su cui e' definita una funzione

$$||\cdot|| : V \rightarrow \mathbb{R}$$

detta ***norma*** che soddisfa le seguenti proprieta':

- $||x|| \geq 0$ e $||x|| = 0$ (*positivita'*)
- $||\alpha x|| = |\alpha| ||x||$ (*omogeneita'*)
- $||x + y|| \geq ||x|| + ||y||$ (*diseguaglianza triangolare*)

inoltre, si dice che uno spazio e' **metrico** se la distanza tra due vettori e' misurata mediante la funzione

$$d(x, y) = ||x - y||$$

Nota: Le norme che verranno utilizzate piu' frequentemente sono la norma a 1, a 2 e a ∞, definite nel modo seguente:

$$||x||_1 = \sum_{i=1}^{n} |x_i|, \quad ||x||_2 = (\sum_{i=1}^{n} x_i^2)^{\frac{1}{2}}, \quad ||x||_\infty = \max_{i=1,\ldots,n} |x_i|$$

Definizione: Una successione si dice di ***Cauchy*** se

$$\lim_{m,n\to\infty} ||x_m - x_n|| = 0$$

6.2 Matrici

Definizione: Il ***determinante*** di una matrice e' una funzione che associa ad ogni ***matrice quadrata*** un numero reale. E' possibile calcolarlo mediante la *formula di Laplace*

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

dove A_{ij} e' la sottomatrice quadrata ottenuta eliminando la riga i-esima e la colonna j-esima. Il determinante di tale sottomatrice e' detto **minore complementare**. Mentre la quantita' $(-1)^{i+j} det(A_{ij})$ e' detto ***cofattore*** o ***complemento algebrico***. In altri termini, il determinante e' la somma dei cofattori moltiplicati per l'elemento nella entrata i, j della matrice A. Il determinante ci dice inoltre se la matrice e' composta interamente da righe e colonne linearmente indipendenti tra di loro. Come significato algebrico, un determinante rappresenta l'area (o volume, dipende dalla dimensione) creata dai vettori che compongono la matrice. Se $= 0$ vuol dire che la trasformazione associata alla matrice, riduce lo spazio di una o piu' dimensioni.

Definizione: Una matrice A e' detta ***invertibile*** o ***non singolare*** se esiste una matrice A^{-1} tale che $AA^{-1} = I = A^{-1}A$. A e' invertibile se e solo se ha $det(A) \neq 0$.
Per calcolare l'inversa di una matrice si calcola la sua matrice dei ***cofattori*** $cof(A)$ corrispondente (guardare sotto per definizione di cofattore). Successivamente si applica la seguente relazione

$$A^{-1} = \frac{1}{det(A)} \cdot cof(A)^T$$

Definizione: Il ***rango*** di una matrice e' il numero massimo di vettori riga (o colonna) linearmente indipendenti che compongono la matrice, oppure come l'ordine della sottomatrice piu' grande con determinante non nullo.

6.3 Autovalori e Autovettori

Definizione: Si dice ***autovalore*** e ***autovettore*** di una matrice A lo scalare λ e il vettore $x \neq 0$ tali che verifichino la relazione

$$Ax = \lambda x$$

Definizione: Il ***polinomio caratteristico*** e' il polinomio $p_A(\lambda)$ costruito come

$$p_A(\lambda) = det(A - \lambda I)$$

La derivazione arriva dal porre $(A - \lambda I)x = 0$, ottenendo un sistema lineare che ha soluzione banale con $x = 0$. Dalla definizione di autovalore e autovettore, pero', bisogna imporre che $x \neq 0$, per cui bisogna rendere a 0 la quantita' $(A-\lambda I)$. Per rendere a 0 tale quantita' si calcola per quali valori ha $det = 0$. Ne deriva che gli zeri (*o radici*) del polinomio caratteristico di A sono gli autovalori di A contati con la loro molteplicita' algebrica. Per il teorema fondamentale dell'algebra, gli zeri del polinomio (e quindi gli autovalori di A) non sono necessariamente reali e distinti tra loro.
Per determinare l'autovettore, si sostituisce in $(A - \lambda_k I)x = 0$, per ogni $\lambda_k \in \{\lambda \mid p_A(\lambda) = 0\}$ autovalore trovato tramite la soluzione del polinomio caratteristico. Si possono trovare cosi' gli autovettori associati all'autovalore λ_k.

Definizione: Lo ***spettro*** di una matrice e' l'insieme di tutti i suoi autovalori

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

Definizione: Il ***raggio spettrale*** di una matrice e' l'autovalore massimo in modulo

$$\rho(A) = \max_{1,\ldots,n} |\lambda_k|$$

Definizione: La ***molteplicita' geometrica*** di un autovalore e' il numero massimo di autovettori linearmente indipendenti associati a tale autovalore. Quando la molteplicita' geometrica e' minore stretta della molteplicita' algebrica la matrice viene detta ***difettiva***.

Alcune proprieta' note di autovalori e autovettori:

- $det(A) = \prod_{k=1}^{n} \lambda_k$
- $\sigma(A^T) = \sigma(A)$, piu' in generale $\sigma(A^p) = \{\lambda_1^p, \ldots, \lambda_k^p\}$
- Ad autovalori distinti corrispondo autovalori linearmente indipendenti
- Il quoziente di ***Rayleigh*** definito come $\frac{x^T Ax}{x^T x}$, fornisce il corrispondente autovalore se e' noto l'autovettore x a priori.

6.4 Matrici di forma particolare

Definizione: Una matrice e' detta ***triangolare superiore*** (analogamente ***triangolare inferiore***) se e' composta da tutti elementi nulli al di sopra (analogamente al di sotto) della diagonale principale. Quando una matrice e' sia diagonale superiore che inferiore e' detta matrice ***diagonale***. Alcune proprieta' importanti riguardanti le matrici triangolari (e diagonali) sono:

- $det(D) = \prod_{i=1}^{n} a_{ii}$ (cioe' il determinante uguale al prodotto degli elementi sulla diagonale principale).
- $\sigma(D) = \{d_{11}, \ldots, d_{kk}\}$, (lo spettro di una matrice coincide con la diagonale principale)

Definizione: Sono dette matrici ***sparse*** le matrici che hanno un numero di elementi diversi da 0 inferiore al 10% del totale degli elementi. Analogamente, una matrice e' detta ***densa*** se la maggior parte dei suoi elementi e' non nulla.

Definizione: Una ***matrice di permutazione*** P e' una matrice che si ottiene permutando le righe della matrice identita' I. L'effetto della matrice di permutazione e' che se viene moltiplicata a sinistra di una matrice ne permuta le righe, mentre se viene moltiplicata a destra ne permuta le colonne.

Definizione: Una matrice quadrata A si dice ***riducibile*** quando esiste una matrice di permutazione P tale che

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

con B_{11} e B_{22} matrici di dimensioni $(k \times k)$ e $(n-k) \times (n-k)$ rispettivamente. Tale proprieta' e' utile per la risoluzione di alcuni sistemi lineari. E' possibile scomporli in sottosistemi composti dalle matrici dei coefficienti B_{11}, B_{12}, B_{22}. Puo' essere anche utile per il calcolo degli autovalori, poiche' e' possibile calolare gli autovalori delle matrici piu' piccole, scomponendo di fatto il problema in due problemi piu' semplici.

Teorema: Sia G il grafo associato alla matrice A, i cui nodi sono costruiti con il numero di righe (o colonne siccome e quadrata), connessi tra loro tramite un arco (i, j), inserito se $a_{ij} \neq 0$. Allora, se G e' ***fortemente connesso*** (cioe' se esiste un cammino orientato per ogni coppia di nodi (i, j)) A e' riducibile.

Definizione: Due matrici A e B si dicono ***simili*** quando esiste una matrice S non singolare $(det(S) \neq 0)$ tale che

$$B = S^{-1}AS$$

Le matrici simili condividono gli stessi autovalori e autovettori.

Definizione: Una matrice A si dice ***diagonalizzabile*** se esiste una matrice X non singolare tale che

$$X^{-1}AX = D = diag(\lambda_1, \ldots, \lambda_n)$$

In altri termini, una matrice diagonalizzabile e' una matrice simile ad una matrice diagonale D. Quando X e' unitaria (cioe' $XX^T = X^TX = I$) allora si dice che A e' ***unitariamente diagonalizzabile***.
Non tutte le matrici sono diagonalizzabili, a tal proposito, diciamo che una matrice A e' diagonalizzabile se e solo se ammette n autovettori linearmente indipendenti. Questo lo si puo' dimostrare applicando la definizione di autovalore e autovettore, siano

$$Ax_i = \lambda_i x_i \quad i = 0, \ldots, n$$

i rispettivi autovalori e autovettori di A, allora possiamo riscrivere la seguente relazione in forma matriciale come

$$A[x_1 \cdots x_n] = [x_1, \cdots, x_n] \begin{bmatrix} \lambda_1 & & \\ & \ddots & \\ & & \lambda_n \end{bmatrix}$$

Ossia $AX = XD$, dove X e' la matrice avente gli autovettori di A come colonne. Se gli autovettori sono indipendenti, la matrice X e' invertibile, per cui si puo' trovare X^{-1} per cui se moltiplicata per la relazione precedente si ottiene

$$X^{-1}AX = D$$

Tale diagonalizzazione e' detta ***fattorizzazione spettrale***, perche' utilizza gli autovalori e gli autovettori di una matrice per ottenere la sua diagonale.

Definizione: Una matrice quadrata A di ordine n e' detta ***diagonalmente dominante*** per righe se

$$a_{ii} \geq \sum_{j=1, j\neq i}^{n} |a_{ij}|, \quad i = 1, \ldots, n$$

Si dice inoltre che e' ***strettamente diagonalmente dominante*** se vale il minore stretto, mentre viene detta ***irriducibilmente diagonalmente dominante*** se e' anche irriducibile.

7 Metodi per la soluzione di sistemi lineari

Un sistema lineare a m equazioni e n incognite e' un insieme di equazioni che devono essere soddisfatte simultaneamente. Una sua soluzione e' una n-upla (tanti elementi quante le incognite) che contiene i numeri reali $\{k_1, \ldots, k_n\}$ tali che soddisfino le equazioni se sostituiti al posto delle incognite.

I sistemi lineari si incontrano molto spesso nella matematica e presentano i seguenti vantaggi:

- E' sempre possibile stabilire a priori se ammettano soluzioni
- In caso ammettano soluzioni e' possibile calcolarle utilizzando un metodo *algebrico* o *numerico*
- Le soluzioni dipendono *esplicitamente dai dati* del problema (coefficienti e termini noti)

Un sistema lineare si presta bene ad essere rappresentato anche in forma matriciale nel modo seguente

$$Ax = b$$

Dove A e' la matrice dei coefficienti, x e' il vettore colonna delle incognite e b e' il vettore colonna dei termini noti. Quando il vettore b e' il vettore nullo 0, si dice che il sistema $Ax = 0$ e' ***omogeneo***. Tali sistemi hanno la particolare caratteristica di avere sempre una soluzione detta *banale* che e' $x = 0$ (vettore con tutte le componenti nulle). I sistemi omogenei possono avere anche altre soluzioni non banali, ma l'esistenza della soluzione nulla e' garantita.

Possiamo anche vedere le soluzioni di un sistema lineare da un punto di vista geometrico

> ***Teorema**: L'insieme delle soluzioni di un sistema omogeneo $Ax = 0$ e' un sottospazio vettoriale $W \subseteq \mathbb{R}^n$, di dimensione $dim(W) = n - r$ in cui $r = rank(A)$.*

L'interpretazione geometrica di un sistema lineare equivale quindi a trovare un vettore x tale che dopo aver applicato la trasformazione A allo spazio diventa uguale a b. Questa relazione deriva dal fatto che il rango di una matrice indica la dimensione del piano dopo aver applicato la trasformazione lineare. (es r=1, allora e' una linea. r=2, e' un piano ecc.). Se abbiamo che $dim(W) = 0$ (perche' la trasformazione non altera il piano in modo significativo riducendone le dimensioni, per cui $rank(A) = n$) abbiamo un solo vettore soluzione, il che e' il caso migliore.

Nella soluzione di sistemi lineari, e' spesso utile poter valutare se tale sistema ammetta o non ammetta soluzioni a priori, prima ancora di effettuare i calcoli effettivi per la risoluzione. In generale e' possibile mediante il seguente teorema

> ***Teorema (Ruche'-Capelli)**: Il sistema*
>
> $$Ax = b$$

dove A e' una matrice di tipo $m \times n$, $x \in \mathbb{R}^n$ e $b \in \mathbb{R}^m$, ammette soluzioni se e solo se

$$rg(A) = rg(A \mid b)$$

dove $[A|b] = [a_1\ a_2 \dots a_n\ b]$ con a_i colonna i-esima di A

Secondo il seguente teorema si possono presentare quindi 3 casi possibili:

- $rank(A) = rank(A \mid b) = n$ *(soluzione unica)*
- $rank(A) = rank(A \mid b) = r < n$ *(infinite soluzioni, sottospazio ∞^{n-r})*
- $rank(A) = rank(A \mid b) \neq n$ *(nessuna soluzione)*

Nei casi dei sistemi omogenei solo la prima o la seconda possono verificarsi. La soluzione unica e' sempre la soluzione banale $x = 0$.

Come detto in precedenza, in presenza di un sistema lineare che ammette soluzioni, e' sempre possibile calcolare la soluzione mediante un opportuno metodo. I metodi numerici per la soluzione di sistemi lineari si suddividono in due categorie principali

- Metodi diretti: In assenza di errori di arrotondamento la soluzione viene calcolata in un numero finito di passi.
- Metodi iterativi: Attraverso un processo iterativo viene generata una successione infinita di vettori che sotto opportune condizioni di convergenza convergono alla soluzione cercata.

La scelta di un algoritmo che si basa su un metodo oppure sull'altro avviene sotto opportuni criteri considerando diversi fattori quali stabilita', complessita' spaziale e computazionale. Tipicamente la scelta viene fatta in funzione delle caratteristiche della matrice dei coefficienti A. Ad esempio, se A e' una matrice *densa*, un algoritmo che sfrutta un metodo diretto potrebbe essere piu' efficiente. Al contrario, in caso A fosse *sparsa*, un metodo diretto potrebbe riempire gli elementi nulli della matrice causando il cosiddetto *"fill-in"*, con un conseguente saturamento della memoria. In questi casi e' spesso utile se non indispensabile impiegare metodi iterativi, che al contrario lasciano inalterata la matrice dei coefficienti.

Prima di introdurre i metodi numerici sono necessarie alcune definizioni fondamentali. In primo luogo definiamo l'errore assoluto e relativo per i vettori.

Definizione (Errore assoluto): *Dato un vettore x non **nullo**, ed un suo vettore approssimante $\tilde{x}$, si dice che $\tilde{x}$ approssima x con n cifre decimali corrette se*

$$||x - \tilde{x}||_\infty \leq \frac{1}{2} \cdot 10^{-n}$$

Definizione (Errore relativo): *Dato un vettore x non **nullo**, ed un suo vettore approssimante $\tilde{x}$, si dice che $\tilde{x}$ approssima x con p*

cifre globali significative corrette se

$$\frac{||x - \tilde{x}||_\infty}{||x||_\infty} \leq \frac{1}{2} \cdot 10^{-p+1}$$

7.0.1 Condizionamento del problema

Oltre che allo studio dell'errore e' fondamentale anche lo studio del condizionamento dei sistemi lineari. In altre parole ci si chiede come un sistema lineare "risponda" ad eventuali perturbazioni nei dati in ingresso (nel caso dei sistemi lineari nei coefficienti e nei termini noti). Supponiamo di avere il sistema lineare $Ax = b$. Per semplicita' ci limitiamo a considerare solo il caso in cui ci sia una perturbazione $\delta b \in \mathbb{R}^n$ sul vettore dei termini noti b. La soluzione del sistema $Ax = (b + \delta b)$ sara' perturbata anch'essa, per cui

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

Di conseguenza, per ottenere δx, sfruttiamo il fatto che $Ax = b$

$$Ax + A\delta x = b + \delta b \ \rightarrow \ A\delta x = \delta b \ \rightarrow \ \delta x = A^{-1}\delta b$$

e, usando $||b|| = ||Ax|| \leq ||A||\ ||x||$ per la compatibilita' della norma si ottiene

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

Notiamo per ora come il fattore $||A||\ ||A^{-1}||$ sia determinante nel condizionamento del sistema perturbato, poiche' amplifica l'errore relativo commesso sul termine noto rispetto all'errore relativo nel calcolo della soluzione.

Prendiamo ora per esempio il caso in cui sia inserita una perturbazione $\delta A \in M_{n,n}(\mathbb{R})$ sulla matrice dei coefficienti A, sempre con i termini noti b perturbati. Secondo le ipotesi si ottiene quindi il seguente sistema perturbato

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

Se supponiamo che $(A + \delta A)$ e' non singolare e che $||A^{-1}||\ ||\delta A|| < \frac{1}{2}$ si ottiene la seguente maggiorazione dell'errore relativo

$$\frac{||\delta x||}{||x||} \leq 2||A||\ ||A^{-1}||(\frac{||\delta A||}{||A||} + \frac{||\delta b||}{||b||})$$

Anche qui compare la quantita' $||A||\ ||A^{-1}||$, detta ***indice o numero di condizionamento*** di A

$$\kappa(A) = ||A||\ ||A^{-1}||$$

E' importante notare come la norma non sia specificata e come tale numero cambi proprio in base alla norma con cui viene valutata. Avendo trovato come le due perturbazioni sui termini noti b e sulla matrice di coefficienti A influenzano le soluzioni, possiamo quindi dare la definizione del seguente teorema

***Teorema**: Siano δA e δb, rispettivamente, le perturbazioni della matrice dei coefficienti e dei termini noti di un sistema lineare $Ax = b$, e sia*

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

il sistema perturbato risultante, se $||\delta A||\ ||A^{-1}|| < 1$, allora l'errore relativo della soluzione del sistema soddisfa la disuguaglianza seguente

$$\frac{||\delta x||}{||x||} \leq \frac{\kappa(A)}{1 - \kappa(A)\frac{||\delta A||}{||A||}}(\frac{||\delta A||}{||A||} + \frac{||\delta b||}{||b||})$$

dove $\kappa(A) = ||A||\ ||A^{-1}||$ e' l'indice di condizionamento di A.

Da cui segue anche il seguente corollario

***Corollario**: Sotto le ipotesi del teorema precedente, se $||\delta A||\ ||A^{-1}|| < \frac{1}{2}$, allora la stima dell'errore relativo della soluzione sara'*

$$\frac{||\delta x||}{||x||} \leq 2\kappa(A)(\frac{||\delta A||}{||A||} + \frac{||\delta b||}{||b||})$$

Dalla stima del corollario precedente e' possibile inoltre ricavare quante cifre sono da ritenersi corrette. Sia ε l'epsilon di macchina si ha

$$\frac{||\delta x||}{||x||} \leq 4\kappa(A)\varepsilon \leq \frac{1}{2}10^{-p+1}$$

per cui p cifre significative sono da ritenersi corrette nella soluzione perturbata.

7.1 Metodi diretti

Dopo aver introdotto alcune nozioni sull'errore e il condizionamento dei sistemi lineari, possiamo passare ad introdurre i metodi per la soluzione dei sistemi lineari, in particolare quelli diretti. Consideriamo inizialmente alcuni casi di sistemi lineari detti "*facili*" perche' di quasi immediata soluzione. Come detto in precedenza, un sistema lineare $Ax = b$ ammette soluzioni se e solo se A e' non singolare (invertibile) e tale soluzione e' data dalla relazione seguente

$$x = A^{-1} \cdot b$$

Se sotto le ipotesi precedenti un sistema ha matrice dei coefficienti diagonale D, il sistema avra' un'unica soluzione

$$x_i = \frac{b_i}{d_i}, \qquad i = 1, \ldots, n$$

Questo caso rappresenta il caso piu' semplice possibile in quanto il costo computazionale e' di n divisioni ($O(n)$), e' ben posto se la matrice e' non singolare ed e' stabile dal momento che ogni divisione e' indipendente l'una dall'altra.

Ricordiamo che *ben posto* significa che un problema possiede in un prefissato campo di definizione una e una sola soluzione e che questa *dipende con continuita' dai dati*.

Consideriamo ora un altro caso relativamente semplice, ipotizzando che la matrice dei coefficienti Q sia ortogonale ($Q^TQ = QQ^T = I$ la trasposta coincide con l'inversa). Poiche' e' certo che Q sia sia invertibile, la soluzione del sistema sara' data da

$$x = Q^T \cdot b \ \rightarrow \ x_i = \sum_{j=1}^{n} q_{ji} b_j, \quad i = 0, \ldots, n$$

Questa volta il costo computazionale coincide con il costo della moltiplicazione di matrici che e' $O(n^2)$ operazioni. Anche in questo caso l'algoritmo e' ben posto e stabile.

Passiamo ora a considerare al caso in cui la matrice dei coefficienti e' triangolare. Definiamo le matrici L e U rispettivamente la matrice triangolare inferiore e superiore. Per la proprieta' delle matrici triangolari sappiamo che i sistemi con matrici triangolari come matrice dei coefficienti ammettono soluzione unica se e solo se il prodotto di tutti gli elementi sulla diagonale principale non e' nullo. In generale i metodi per la risoluzione di sistemi triangolari si basano sul concetto di sostituzione. Prendiamo per esempio un sistema triangolare inferiore $Lx = b$ (il concetto viene esteso anche al caso di sistemi triangolari superiori), allora possiamo scrivere lo stesso in forma esplicita come:

$$\begin{cases} l_{1,1}x_1 & = b_1 \\ l_{2,1}x_1 + l_{2,2}x_2 & = b_2 \\ l_{3,1}x_1 + l_{2,2}x_2 + l_{3,3}x_3 & = b_3 \\ \quad \vdots & \quad \vdots \\ l_{3,1}x_1 + l_{2,2}x_2 + l_{3,3}x_3 + \cdots + l_{n,n}x_n & = b_n \end{cases}$$

E' evidente dalla forma esplicita come sia possibile ricavare immediatamente la prima incognita x_1 dalla prima equazione, per poi generare cosi' un *"effetto a cascata"* sostituendo il valore di x_1 in tutte le altre equazioni e ripetendo il procedimento nella riga successiva. Cio' che e' appena stato descritto e' in effetti il ***metodo della sostituzione in avanti*** o ***forward substitution***. Posto che $x_1 = \frac{b_1}{l_{1,1}}$, allora possiamo calcolare il generico x_i come

$$x_i = \frac{b_i - \sum_{k=1}^{i-1} l_{i,k}x_k}{l_{i,i}}, \quad i = 2, \ldots, n$$

Siccome abbiamo posto che la matrice non sia singolare e quindi non ci siano elementi nulli sulla diagonale principale, il termine $l_{i,i}$ non causa problemi di stabilita'.

L'algoritmo puo' essere riassunto nei passi seguenti in codice Matlab:

```
1  x1 = b1/L(1  1);
2  for  i=2:n
3       x(i) = b(i);
4       for  k=1:i-1
5            x(i) = x(i) - L(i  k) * x(k);
6       end
7       x(i) = x(i) / L(i  i);
8  end
```

Valutando l'algoritmo si ottiene che la sua complessita' e di

$$O(\sum_{i=1}^{n} i) = O(\frac{n(n+1)}{2}) = O(\frac{n^2}{2})$$

Come gia' detto possiamo applicare lo stesso procedimento al contrario, ottenendo il cosiddetto ***metodo della sostituzione in indietro*** o ***backward substitution***. Anche l'algoritmo di backward substitution gode delle stesse caratteristiche della sua controparte quali stabilita' e costo computazionale. In generale, dato $x_n = \frac{b_n}{l_{n,n}}$ allora la generica componente x_i del vettore delle soluzioni puo' essere calcolata mediante sostituzione in indietro per mezzo della seguente relazione

$$x_i = \frac{b_i - \sum_{k=i+1}^{n} u_{i,k} x_k}{u_{i,i}}, \quad i = (n-2), (n-3), \ldots, 1$$

7.2 Metodo di Gauss

I metodi di risoluzione fin'ora introdotti si limitano ad alcuni casi limite e molto particolari. Introduciamo ora alcuni metodi per la risoluzione di sistemi lineari di forma generica, con la matrice dei coefficienti senza nessuna forma particolare. Il metodo di Gauss e' un metodo che consiste nell'eliminare progressivamente le incognite da fissate equazioni. Tale metodo funziona sempre, sia se il sistema non ammette una sola soluzione sia nel caso ne ammetta infinite. Nel secondo caso l'algoritmo termina segnalandolo. L'idea alla base del metodo e' quella di ottenere mediante un numero finito di passi una matrice dei coefficienti triangolare. Dopo aver triangolarizzato la matrice e' poi possibile utilizzare uno dei metodi di sostituzione descritti in precedenza.

Assumiamo di avere un sistema lineare $Ax = b$, e assumiamo anche che la prima entrata $a_{1,1}$ di A sia diversa da zero ($a_{1,1} \neq 0$). Possiamo quindi eliminare la prima incognita x_1 dalla $2^a, 3^a, \ldots,$ *n-esima* equazione, sottraendo dall'*i-esima* equazione ($i = 2, \ldots, n$) la prima equazione moltiplicata per

$$m_{i,1} = \frac{a_{i,1}}{a_{1,1}}, \quad i = 2, \ldots, n$$

Cio' che si ottiene e' un sistema equivalente, con la prima equazione inalterata ma con tutti i coefficienti di x_1 uguali a 0 in tutte le equazioni successive.

Piu' precisamente, il sistema lineare equivalente (il cui passo dell'algoritmo e' specificato ad apice, se non specificato e' passo 1) sara'

$$\begin{cases} a_{i,j}^{(2)} = a_{i,j} - m_{i,1}a_{1,j} \\ \\ b_i^{(2)} = b_i - m_{i,1}b_1 \end{cases} \qquad i,j = 2,\ldots,n$$

Il metodo viene poi reiterato prendendo $a_{2,2}^{(2)}$ come elemento per l'eliminazione, ottenendo quindi un sistema in cui le due prime equazioni sono inalterate, ma con i coefficienti di x_1 e x_2 uguali a zero. Per cui il sistema risultante sara':

$$\begin{cases} a_{i,j}^{(3)} = a_{i,j}^{(2)} - m_{i,2}a_{2,j}^{(2)} \\ \\ b_i^{(3)} = b_i^{(2)} - m_{i,2}b_2^{(2)} \end{cases} \qquad i,j = 3,\ldots,n$$

E' chiaro fin da subito che il metodo possa essere reiterato per il passo *k-esimo*, ottenendo alla fine un sistema con matrice dei coefficienti triangolare superiore. Per il caso generale, possiamo riassumere il metodo con un insieme di relazioni. Posto che

$$a_{i,j} = a_{i,j}^{(1)}, \quad b_i = b_i^{(1)}, \quad i,j = 1,\ldots,n$$

allora e' possibile calcolare il sistema al passo $k+1$ mediante le seguenti equazioni:

$$\begin{cases} m_{i,k} = \frac{a_{i,k}^{(k)}}{a_{k,k}^{(k)}} \\ \\ a_{i,j}^{(k+1)} = a_{i,j}^{(k)} - m_{i,k}a_{k,j}^{(k)} \\ \\ b_i^{(k+1)} = b_i^{(k)} - m_{i,k}b_k^{(k)} \end{cases} \qquad i,j = k+1,\ldots,n$$

L'elemento $a_{k,k}$ e' detto ***elemento pivot***, mentre le quantita' $m_{i,k}$ sono dette ***moltiplicatori***. Dal metodo si evince inoltre che l'algoritmo e' stabile se e solo se tutti gli elementi pivot sono diversi da 0 (dal momento che compaiono al denominatore). L'algoritmo termina inoltre in $n-1$ passi, con n dimensione della matrice quadrata dei coefficienti.

Algoritmo di Gauss:

```
1  for k=1:n-1
2      for i=k+1:n
3          m(i k) = a(i k)/a(k k);
4          for j=k+1:n
5              a(i j)=a(i j)-m(i k)*a(k j);
6          end
7          b(i)=b(i)-m(i k)*b(k);1
8      end
9  end
```

Dal codice si evince che la complessita' temporale e' di tipo $O(n^3)$

7.2.1 Stabilita'

La stabilita' del metodo di Gauss e' garantita in generale se tutti se per ogni passo k dell'algoritmo gli elementi pivot $a_{k,k}$ sono diversi da 0. Per poterlo stabilire a priori si puo' utilizzare il seguente teorema:

> ***Teorema**: Sia $A \in M_{n,n}(\mathbb{R})$. Gli elementi pivot sono tutti diversi da zero se e soltanto se tutte le matrici prncipali di testa sono invertibili, (o non singolari) quindi se*
>
> $$det(A_k) \neq 0 \quad \forall k = 1, \ldots, n$$
>
> *dove le matrici principali di testa A_k sono definite nel modo seguente*
>
> $$A_k(a_{i,j})_{i,j=1,\ldots,k}, \quad k = 1, \ldots, n$$

Dal teorema precedente e' evidente come non sia una condizione di facile verifica e chiaramente computazionalmente molto onerosa da verificare. Tuttavia, esistono alcune classi di matrici per cui e' noto che tutti gli elementi pivot siano diversi da zero per ogni k, quali:

- Matrici diagonalmente dominanti per righe
- Matrici diagonalmente dominanti per colonne
- Matrici simmetrice definite positive

La terza tipologia infatti si basa sul risultato del seguente teorema

> ***Teorema (Criterio di Sylvester)**: Una matrice simmetrica $A \in M_{n,n}(\mathbb{R})$ e' definita positiva se e solo se*
>
> $$det(A_k) > 0, \quad k = 1, \ldots, n$$
>
> *dove $det(A_k)$ sono i **minori principali** di A.*

In generale, pero', per ogni matrice A non singolare (condizione necessaria per la soluzione di un sistema lineare) non si ha la certezza che ogni elemento pivot ad ogni passo k non sia nullo. Tuttavia, quando l'elemento pivot al generico passo k e' nullo, e' impossibile che non esista una riga della matrice dei coefficienti che abbia pivot diverso da 0, questo perche' se fosse cosi', allora la matrice dei coefficienti sarebbe singolare e non si potrebbe di fatto risolvere il sistema. L'idea e' quindi quella di scambiare la riga *k-esima* con pivot uguale a 0 con la riga *r-esima* che ha pivot non nullo. Tale tecnica di scambio di righe e' detta **pivoting**. Siccome al pari di scambi di righe il sistema lineare rimane invariato, possiamo affermare che

> *Ogni sistema non singolare, mediante opportuni scambi di righe, puo' essere sempre ricondotto alla forma triangolare superiore con il metodo di Gauss.*

E' possibile inoltre utilizzare il pivoting anche quando i pivot sono molto piccoli (in valore assoluto) rispetto all'ordine di grandezza degli elementi della matrice. Questo perche' potrebbe causare il fenomeno della *cancellazione numerica*, e quindi rappresenterebbe fonte di *instabilita'*.
Otteniamo cosi' due strategie principali di pivoting:

Pivoting Parziale (o di colonna): cerca il nuovo pivot considerando tutti gli elementi della sottocolonna *k-esima*, (che hanno riga $l \geq k$) scegliendo il massimo.

- Al passo k:
  1. Trova l tale che $|a_{l,k}^{(k)}| = \max_{i=k,\dots,n} |a_{i,k}^{(k)}|$
  2. Scambia la riga k con la riga l
  3. Scambia gli elementi k e l di b

Pivoting Totale: cerca il nuovo pivot considerando l'intera sottomatrice di ordine $n-k$, scegliendo il massimo.

- Al passo k:
  1. Trova (r,s) con $r,s \geq k$ tale che

$$|a_{r,s}^{(k)}| = \max_{k \leq i,j \leq n} |a_{i,j}^{(k)}|$$

  2. Scambia le righe k e r e le colonne k e s
  3. Scambia gli elementi k e r di b
  4. Memorizza che sono state scambiate le incognite x_k x_s

In generale, la strategia di pivoting totale e' molto costosa computazionalmente, per questo motivo si tende ad impiegare piu' spesso la strategia di pivoting parziale, anche perche' risulta soddisfacente nella maggior parte dei casi.
Osserviamo inoltre che nei casi di matrici diagonalmente dominanti per colonne, il pivoting non genera scambi, mentre nei casi di matrici simmetriche positive il pivoting produce scambi ma senza apportare nessun miglioramento significativo. Si dice percio' che in questi casi la stabilita' e' garantita anche senza pivoting.

7.2.2 Condizionamento

Avendo discusso della stabilita', si vuole a questo punto studiare la propagazione degli errori del metodo di Gauss. A tal proposito, enunciamo il seguente teorema

> ***Teorema (Wilkinson):*** *La soluzione numerica $\tilde{x}$ del sistema lineare $Ax = b$ di ordine n, ottenuta mediante l'applicazione del metodo di Gauss con pivoting (parziale o totale) in un sistema a virgola mobile in base β con t cifre significative, coincide con la soluzione esatta del sistema perturbato*
>
> $$(A + \delta A)\tilde{x} = b$$
>
> *Se $n^2\beta^{-1} << 1$, (cioe' che lo spazio destinato del sistema in virgola mobile alla memorizzazione di un numero reale sia sufficientemente*

grande rispetto alla dimensione del sistema) allora si ha

$$||\delta A||_\infty \leq 2g(n)\beta^{-t}(n+1)^3||A||_\infty$$

dove il fattore di crescita $g(n)$ assume i valori

$$\begin{cases} 2^{n-1} & \text{per il pivoting parziale} \\ n^{\frac{1}{2}}(2 \cdot 3^{\frac{1}{2}} \cdot 4^{\frac{1}{3}} \dots n^{\frac{1}{(n-1)}})^{\frac{1}{2}} & \text{per il pivoting totale} \end{cases}$$

Ora applichiamo la definizione di errore relativo data dal teorema illustrato nella sezione 7.0.1, e otteniamo:

$$\frac{||\delta x||}{||x||} = \frac{||x - \tilde{x}||}{||x||} \leq \frac{||A||\,||A^{-1}||}{1 - ||A||\,||A^{-1}||\,\frac{||\delta A||}{||A||}} \cdot \frac{||\delta A||}{||A||} = \frac{||A^{-1}||\,||\delta A||}{1 - ||A^{-1}||\,||\delta A||}$$

il che ci dice che l'errore relativo tende a 0 quando la norma della perturbazione sui dati della matrice dei coefficienti δA tende a 0.

Osservazione: C'e' da dire che anche se esistono matrici il cui fattore di crescita relativo al pivoting parziale raggiunge il suo limite superiore di 2^{n-1}, in molti casi i fattori di crescita sono molto piu' bassi del massimo (ne sono quindi un limite inferiore). In questo caso le due strategie di pivoting producono risultati molto simili in termini di errore. In generale, e' consigliabile fare ricorso al pivoting totale in casi in cui ci siano sistemi con matrici dei coefficienti di grandi dimensioni.

7.3 Fattorizzazione LU

La fattorizzazione LU e' un metodo strettamente legato al metodo di Gauss. Consideriamo una matrice quadrata A di ordine n ($A \in M_{n,n}(\mathbb{R})$), tramite il metodo di Gauss e' possibile rappresentare A come

$$A = LU$$

dove L e' una matrice triangolare inferiore a diagonale unitaria e U una matrice triangolare superiore definite nel modo seguente

$$L = \begin{bmatrix} 1 & 0 & 0 & \cdots & 0 \\ m_{2,1} & 1 & 0 & \cdots & 0 \\ m_{3,1} & m_{3,2} & 1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ m_{n,1} & m_{n,2} & \cdots & m_{n,n-1} & 1 \end{bmatrix} \quad U = \begin{bmatrix} a_{1,1}^{(1)} & a_{1,2}^{(1)} & \cdots & \cdots & a_{1,n}^{(1)} \\ 0 & a_{2,2}^{(2)} & \cdots & \cdots & a_{2,n}^{(2)} \\ 0 & 0 & a_{1,3}^{(3)} & \cdots & a_{3,n}^{(3)} \\ \cdots & \ddots & \ddots & \ddots & \vdots \\ 0 & \cdots & 0 & \cdots & a_{n,n}^{(3)} \end{bmatrix}$$

Come nel caso del MEG, la fattorizzazione LU esiste ed e unica se e solo se la matrice A e' non singolare e tutti i suoi minori principali sono non nulli. Il costo computazionale della fattorizzazione LU e' $O\left(\frac{n^3}{3}\right)$. Questo perche' la

fattorizzazione LU coincide con la prima parte del metodo di Gauss, tralasciando quindi la parte in cui si fa una sostituzione in avanti. (non si vuole risolvere un sistema lineare ma si vuole ottenere solo una fattorizzazione della matrice A)

Nonostante cio', mediante la fattorizzazione LU e' anche possibile risolvere sistemi lineari, e lo si puo' fare procedendo nel modo seguente:

1. Calcolati L e U, si ottiene che $Ax = b \leftrightarrow LUx = b$
2. Posto che $Ux = y$, si calcola la soluzione del sistema $Ly = b$ mediante *forward substitution.*
3. y diventa cosi' il termine noto del sistema triangolare superiore $Ux = y$, che puo' essere risolto mediante *backward substitution.* Tale soluzione e' la soluzione del sistema iniziale.

Il costo computazionale della procedura e' superiore a quella del metodo di Gauss poiche' risulta essere di $\frac{n^3}{3} + 2\frac{n^2}{2}$ (applico Gauss + faccio 2 sostituzioni -avanti e indietro- che hanno la stessa complessita') flops. A differenza dell'algoritmo di Gauss, la fattorizzazione LU non modifica i termini noti del sistema. LU quindi opera solamente sulla matrice dei coefficienti, lasciando invariato il vettore dei termini noti.

Concettualmente, quindi, LU e' semplicemente il metodo di Gauss in cui pero' si salvano i moltiplicatori di ogni passo nella matrice triangolare inferiore L. La fattorizzazione LU e' inoltre particolarmente utile per diverse ragioni tra cui:

- **Efficiente occupazione della memoria**: presenta la minore complessita' spaziale ($O(n(n+1))$) possibile, che coincide con quella necessaria a memorizzare i dati del problema.
- **Risoluzione di piu' sistemi lineari**: dal momento che lascia invariati gli altri dati del sistema, e' particolarmente utile per la soluzione di m sistemi lineari aventi la stessa matrice dei coefficienti ma vettori dei termini noti differenti. Il costo computazionale per la soluzione di m sistemi, ciascuno risolto indipendentemente dagli altri e' di $O\left(\frac{n^3}{3} + \frac{n^2}{2}\right)$, mentre se prima si opera la fattorizzazione LU e poi si ripetono solo i passi 2) e 3) descritti in precedenza si ottiene una complessita' pari a $O\left(\frac{n^3}{3} + mn\right)$
- **Calcolo del determinante**: per il teorema di Binet abbiamo che $det(A) = det(LU) = det(L) \cdot det(U)$. Dal momento che $det(L) = 1$, allora abbiamo che $det(A) = det(U) = \prod_{i=1}^{n} u_{i,i}$
- **Calcolo dell'inversa di una matrice**: dal momento che A e' invertibile, le colonne della matrice inversa di A saranno le soluzioni dei sistemi di equazioni che hanno A come matrice dei coefficienti e il vettore e^n della base canonica di $\mathbb{R}^n$. In altri termini, sono le colonne *n-esime* della matrice identita' di ordine n. Il costo computazionale e' pari a $O\left(\frac{4}{3}n^3\right)$

La strategia di pivoting discussa in precedenza e' applicabile anche alla fattorizzazione LU. Sia $P^{(k,s)}$ la matrice di permutazione ottenuta scambiando la *k-esima* e *s-esima* riga tra loro. Si puo' dimostrare che la fattorizzazione LU con

pivoting parziale produce la fattorizzazione

$$PA = LU$$

Dove P e' la matrice di permutazione in cui vengono inserite tutte gli scambi che vengono effettuati dalla strategia di pivoting ad ogni passo

$$P = P^{(k_m, s_m)} \cdot P^{(k_{m-1}, s_{m-1})} \dots P^{(k_1, s_1)}$$

Si puo' successivamente procedere con la soluzione di sistemi nel modo precedente, facendo solo qualche accortezza, dal momento che bisogna tener conto della permutazione anche nel vettore dei termini noti b:

$$Ax = b \leftrightarrow PAx = Pb \leftrightarrow LUx = Pb$$

e di conseguenza

$$det(A) = (-1)^{\#(P)} \prod_{i=1}^{n} u_{i,i}$$

8 Localizzazione degli Autovalori

8.1 Funzioni lineari

In algebra lineare, una funzione lineare e' una funzione che associa ad un vettore $x \in \mathbb{R}^n$ il vettore $f(x) \in \mathbb{R}^m$ che gode della proprieta' di linearita'

$$\forall x, y \in \mathbb{R}^n \text{ e } \forall \alpha, \beta \in \mathbb{R} \qquad f(\alpha x + \beta y) = \alpha f(x) + \beta f(y)$$

In altre parole, e' una funzione che preserva le operazioni di somma tra vettori e prodotto per uno scalare. Per esempio, la funzione $f : \mathbb{R}^3 \to \mathbb{R}$, che associa al vettore $x \in mathbbR^3$ il numero reale:

$$f(x) = 2x_1 - 4x_2 + 7x_3$$

e' lineare, in quanto soddisfa la proprieta' di linearita' precedente. Infatti, $\forall x, y \in \mathbb{R}^3$ e $\forall \alpha \in \mathbb{R}$ si ha:

$$f(x+y) = 2(x_1 + y_1) - 4(x_2 + y_2) + 7(x_3 + y_3) = 2x_1 - 4x_2 + 7x_3 + 2y_1 - 4y_1 + 7y_3 = f(x) + f(y)$$
$$f(\alpha x) = 2(\alpha x_1) - 4(\alpha x_2) + 7(\alpha x_3) = \alpha(2x_1 - 4x_2 + 7x_3) = \alpha f(x)$$

Dal momento che moltiplicando una matrice $A \in \mathbb{R}^{(m,n)}$ per un vettore $x \in \mathbb{R}^n$ si ottiene un vettore $y \in \mathbb{R}^m$, possiamo rappresentare una qualsiasi applicazione lineare semplicemente mediante una matrice A. Data quindi una matrice A e' possibile costruire un'applicazione lineare f equivalente

$$f(x) = A \cdot x$$

Sappiamo inoltre che e' lineare poiche' segue dalla proprieta' del prodotto fra matrici.

> ***Teorema**: Una funzione $f : \mathbb{R}^n \to \mathbb{R}^m$ e' lineare se e solo se esiste una matrice A di tipo $m \times n$ tale che:*
>
> $$f(x) = A \cdot x$$
>
> *inoltre la matrice A e' unica una volta fissate le basi $\mathbb{R}^n$ e $\mathbb{R}^m$.*

Per il teorema precedente possiamo quindi ragionare sulle trasformazioni lineari in modo puramente algebrico, senza dover utilizzare strumenti tipici dell'analisi matematica.

8.2 Autovalori e autovettori

Ragioniamo ora sempre in termini di matrici come trasformazioni lineari, e diamo la definizione di autovalore e autovettore in questi termini

> ***Definizione:** Sia $f : \mathbb{R}^n \to \mathbb{R}^n$ una trasformazione lineare. Se $v \in \mathbb{R}^n, v \neq 0$, e $\lambda \in \mathbb{R}$, sono tali che*
>
> $$f(x) = \lambda v$$

Allora v e' autovettore della ***trasformazione*** *f e λ e' il suo* ***autovalore***

Dalla proprieta' di linearita' della trasformazione f, abbiamo che se v e' un autovettore di f con autovalore λ, allora qualsiasi altro multiplo di $v \neq 0$ sara' anch'esso un autovettore con lo stesso autovalore λ. Si dice infatti che gli autovettori aventi lo stesso autovalore fissato λ, generano un sottospazio vettoriale chiamato *autospazio* relativo all'autovettore λ.

In generale, possiamo parlare di autovettori e autovalori anche in termini della matrice di trasformazione A associata a f, per cui abbiamo che λ e v sono rispettivamente autovalori e autovettori della matrice A se

$$Av = \lambda v$$

Richiamiamo brevemente le definizioni viste in precedenza. Abbiamo che per calcolare gli autovalori si ricorre al polinomio caratteristico $P_n(\lambda) = det(A - \lambda I_n)$, mentre per calcolare gli autovettori si risolve il sistema omogeneo $(A - \lambda I_n)x = 0$ rispetto a x. In caso siano gia' noti gli autovalori λ e' possibile ottenere gli autovettori tramite il quoziente di Rayleigh.

8.3 Localizzazione degli autovalori

Tramite alcuni teoremi e' possibile localizzare gli autovalori di una matrice all'interno del piano complesso. Il primo di questi teoremi (e anche il piu' generico) e' il seguente

Teorema (Hirsch)*: Sia A una matrice di ordine n e sia $||\cdot||$ una qualsiasi delle tre norme di matrici tra $\{1, 2, \infty\}$. Allora il cerchio definito come*

$$\{z \in \mathbb{C} \ : \ |z| \leq ||A||\}$$

contiene tutti gli autovalori di A.

Vediamo ora pero' un teorema piu' raffinato rispetto al teorema precedente, in quanto ci da informazioni piu' precise per la localizzazione degli autovalori nel piano complesso. Il teorema e' il seguente:

Primo Teorema di Gerschgorin*: Gli autovalori di una matrice A di ordine n sono contenuti nell'insieme*

$$S_R = \bigcup_{i=1}^{n} R_i$$

dove

$$R_i = \left\{ z \in \mathbb{C} \ : \ |z - a_{ii}| \leq \sum_{j=1, j \neq i}^{n} |a_{ij}| \right\}$$

con $i = 1, \ldots, n$ sono detti ***cerchi riga di Gerschgorin****.*

Cioe' un cerchio riga corrisponde ad un cerchio di centro a_{ii} e raggio corrispondente alla somma di tutti gli elementi sulla riga corrente, eccetto l'elemento a_{ii}.

Dimostrazione: Sia λ un autovalore di A e x un autovettore corrispondente. Nella definizione di autovalore e autovettore, e' possibile riscrivere la relazione esplicitando il prodotto riga per colonna nel modo seguente

$$Ax = \lambda x \quad \leftrightarrow \quad \sum_{j=1}^{n} a_{ij}x_j = \lambda x_i \qquad i = 1, \ldots, n$$

Sia x_p tale che $|x_p| = \max_{j=1,\ldots,n} |x_j|$, allora possiamo *"portare fuori"* dalla somma il termine x_p, e considero la relazione precedente per $i = p$

$$\sum_{j\neq p}^{n} a_{pj}x_j + (a_{pp}x_p) = \lambda x_p \quad \leftrightarrow \quad (\lambda - a_{pp})x_p = \sum_{j\neq p}^{n} a_{pj}x_j$$

e quindi, per la proprieta' del modulo e per il fatto che x_p e' l'elemento di modulo massimo, possiamo riscrivere la relazione precedente come

$$|\lambda - a_{pp}|\,|x_p| \leq \sum_{j\neq p}^{n} |a_{pj}|\,|x_j| \leq \sum_{j\neq p}^{n} |a_{pj}|\,|x_p|$$

Dividendo infine per $|x_p| \neq 0$, dal momento che x e' un autovalore, per la definizione non puo' essere un vettore nullo, per cui esistera' per forza un elemento $\neq 0$. Dividendo per questa quantita' otteniamo

$$|\lambda - a_{pp}| \leq \sum_{j\neq p}^{n} |a_{pj}|.$$

Poiche' a priori non e' noto il valore di p, possiamo dedurre solo che $\lambda \in \bigcup_{i=1}^{n} R_i$. Naturalmente questo discorso possiamo estenderlo a tutti gli autovalori della matrice A, ottenedo quindi la tesi ■.

Dal momento che il teorema puo' essere applicato anche ad A^T, possiamo anche definire i **cerchi colonna di Gerschgorin** C_i. Quindi, per avere una localizzazione piu' precisa possiamo combinare sia i cerchi riga che i cerchi colonna e affermare che gli autovalori di A sono contenuti nell'insieme

$$S_R \cap S_C$$

dove

$$S_R = \bigcup_{i=1}^{n} R_i \quad \text{e} \quad S_C = \bigcup_{i=1}^{n} C_i$$

Vediamo ora il secondo e terzo teorema di Gerschgorin

***Secondo Teorema di Gerschgorin**: Se l'unione S_1 dei k cerchi di Gerschgorin e' disgiunta dall'unione S_2 dei rimanenti $n-k$, cioe'*

$$S_1 = \bigcup_{i=1}^{k} R_i, \qquad S_2 = \bigcup_{i=k+1}^{n} R_i, \qquad S_1 \cap S_2 = \emptyset$$

allora S_1 contiene k autovalori di A e S_2 i restanti $n-k$, contati con la loro molteplicita'.

Per esempio, se un cerchio e' disgiunto da tutti gli altri, allora contiene un solo autovalore ($k = 1$). Inoltre sappiamo che tale valore e' reale (altrimenti anche il suo coniugato dovrebbe appartenere al cerchio). Se invece ad esempio tutti i cerchi sono disgiunti, vuol dire che tutti gli autovalori di A sono reali con molteplicita' 1.

Introduciamo per finire il terzo teorema di Gerschgorin e un lemma importante che ne deriva

***Terzo Teorema di Gerschgorin**: Sia A una matrice irriducibile. Se un suo autovalore appartiene alla frontiera S_R, esso deve appartenere allla frontiera di ciascun cerchio R_i, con $i = 1, \ldots, n$.*

Possiamo ora combinare i teoremi precedenti per dimostrare un importante teorema:

***Teorema della dominanza diagonale:** Se A e' strettamente o irriducibilmente diagonalmente dominante, allora e' non singolare.*

Da cui ne deriva il seguente

***Corollario**: Se e' verificata una delle ipotesi del teorema precedente e se la matrice A e' hermitiana (nel caso reale simmetrica) con $a_{ii} > 0, i = 0, \ldots, n$, allora A e' definita positiva.*

Dimostrazione: Poiche' A e' hermitiana (nel caso reale simmetrica), i suo autovalori sono reali. Inoltre, le ipotesi che $a_{ii} > 0$, la dominanza diagonale e il primo teorema di Gerschgorin implicano che $\lambda_i \geq 0, i = 0, \ldots, n$.
Infine, essendo la matrice non singolare per il teorema della dominanza diagonale, si ha che $\lambda_i > 0, i = 1, \ldots, n$ ■.

Questo corollario ci chiarisce del perche' il metodo di Gauss senza pivoting sia stabile per matrici diagonalmente dominanti.

8.4 Esercizi

Esercizio 1: Si applichino il teorema di Hirsch e i teoremi di Gerschgorin per localizzare gli autovalori della matrice

$$A = \begin{bmatrix} 15 & -2 & 2 \\ 1 & 10 & -3 \\ -2 & 1 & 0 \end{bmatrix}$$

Soluzione:

Esercizio 2: Utilizzare i teoremi di Gerschgorin per localizzare gli autovalori e rispondere ai seguenti quesiti.

- La matrice A e' diagonalizzabile?
- La matrice B ha qualche autovalore non reale?
- La matrice C e' invertibile? Ha un autovalore > 20?

con

$$A = \begin{bmatrix} 1 & 0.1 & 0.2 \\ 0.2 & 4 & 0.3 \\ 0.4 & 0.5 & 8 \end{bmatrix}, \quad B = \begin{bmatrix} 17 & 2 & 2 \\ 1 & 10 & 3 \\ 1 & 2 & 0 \end{bmatrix} \quad C = \begin{bmatrix} 3 & 0 & 1 & 0 \\ 0 & 11 & 1 & -1 \\ 1 & 1 & 10 & 2 \\ 0 & -1 & 2 & 20 \end{bmatrix}$$

Soluzione:

9 Metodi numerici per il calcolo degli autovalori

9.1 Condizionamento

Fino ad ora abbiamo studiato la teoria concernente gli autovalori di matrici. In particolare abbiamo trattato diversi importanti teoremi che ci permettono di localizzarli all'interno del piano complesso. In questo capitolo si vogliono studiare invece dei metodi numerici per il calcolo degli autovalori. Prima di introdurre direttamente dei metodi, pero', e' importante prima valutare e studiare il condizionamento del problema. In particolare, il seguente teorema ci da' delle informazioni dirette sul condizionamento del problema del calcolo degli autovalori in modo numerico

> ***Teorema (Bauer-Fike)**: Sia $|| \cdot ||$ una norma **assoluta**, e sia $A \in M_{n,n}(\mathbb{R})$ una matrice diagonalizzabile con X matrice che ha per colonne gli autovettori di A. Se δA e' la matrice delle perturbazioni indotte su A e μ e' un autovalore di $A + \delta A$, allora esiste almeno un autovalore λ di A tale che*
>
> $$|\lambda - \mu| \leq k(X)\ ||\delta A||$$
>
> *dove $k(X) = ||X||\ ||X^{-1}||$ e' l'indice di condizionamento di X.*

Nota: Una norma matriciale indotta $|| \cdot ||$ si dice **assoluta** quando

$$||D|| = \max_{i=0,\dots,n} |d_{ii}|$$

per ogni matrice diagonale $D \in M_{n,n}(\mathbb{R})$. Le norme 1, 2, ∞ sono norme assolute.

Dimostrazione: Escludiamo per prima cosa il caso banale, in cui se μ fosse un autovalore di A, allora l'errore sara' nullo, per cui vale $0 \leq k(X)||\delta A||$. Consideriamo ora il caso in cui μ non sia un autovalore di A. In questo caso, $A - \mu I$ e' non singolare, per cui con y autovettore corrispondente all'autovalore μ e per la definizione di autovalore e autovettore otteniamo

$$(A + \delta A)y = \mu y \ \leftrightarrow\ (A - \mu I)y = -\delta A y \ \leftrightarrow\ y = -(A - \mu I)^{-1}\delta A y$$

Applicando la consistenza delle norme ($||Ax|| \leq ||A|| \cdot ||x||$) ed essendo $||y|| \neq 0$, e' possibile dividere tutto per $||y||$, ottenendo

$$||y|| \leq ||(A - \mu I)^{-1}\delta A y|| \leq ||(A - \mu I)^{-1}\delta A||\ ||y|| \ \rightarrow\ 1 \leq ||(A - \mu I)^{-1}\delta A||.$$

Sfruttiamo ora la submoltiplicativita' della norma matriciale ($||AB|| \leq ||A|| \cdot ||B||\$$) e l'ipotesi che A sia diagonalizzabile ($A = XDX^{-1}$), ottenendo che

$$(A - \mu I)^{-1} = (XDX^{-1} - \mu XDX^{-1})^{-1} = X(D - \mu I)^{-1}X^{-1}$$

Sostituiamo ora la relazione trovata, per cui

$$1 \leq ||X(D - \mu I)^{-1}X^{-1}\delta A|| \neq ||X||\ ||(D - \mu I)^{-1}||\ ||X^{-1}||\ ||\delta A||.$$

Essendo la norma assoluta, vale quindi che

$$||(D-\mu I)^{-1}|| = \max_i |\frac{1}{\lambda_i - \mu}| = \frac{1}{\min_i |\lambda_i - \mu|}$$

e quindi, infine

$$\min_i |\lambda_i - \mu| \leq ||X||\ ||X^{-1}||\ ||\delta A|| = k(X)||\delta A||$$

■

***Corollario**: Se $A \in M_{n,n}$ e' **normale** ($A^T \cdot A = A \cdot A^T$), allora*

$$|\lambda - \mu| \leq ||\delta A||_2$$

Dimostrazione: Se A e' normale, allora X sara' per forza ortogonale, per cui $X \cdot X^T = I = X^T \cdot X$, di conseguenza

$$||X||_2 = ||X^T||_2 = \sqrt{\rho(X^T X)} = \sqrt{\rho(I)} = 1$$

e quindi, siccome $||X||_2 = 1$, la stima del fattore di condizionamento $k(X)$ sara'

$$k_2(X) = ||X||_2\ ||X^T||_2 = 1$$

Sostituendo la stima $k_2(X)$ all'interno della stima data dal teorema di Bauer-Fike si ottiene la tesi. ■

Questo corollario ci dice quindi che per matrici normali il calcolo di tutti gli autovalori e' ben condizionato. Questo ci dice anche che per matrici dei coefficienti che sono mal condizionate per la soluzione di sistemi lineari, potrebbero non essere mal condizionate nel calcolo degli autovalori.

Per matrici generiche, sappiamo pero' che l'errore nel calcolo degli autovalori dipende proporzionalmente dall'entita' delle perturbazioni della matrice δA. Il teorema di Bauer-Fike ci dice anche che tale errore e' influenzato direttamente anche dall'indice di condizionamento della matrice degli autovalori X. L'errore sara' quindi tanto piu' basso quanto piu' basso sra' $k(X)$.

Siccome il problema del calcolo degli autovalori potrebbe essere ben condizionato o mal condizionato a seconda della molteplicita' algebrica degli autovalori considerati, e' bene distinguere i casi differenti tra di loro. Consideriamo inzialmente il condizionamento del calcolo di autovalori di molteplicita' algebrica pari a 1:

> ***Teorema**: Sia $A \in M_{n,n}$ una matrice diagonalizzabile con X matrice che ha per colonne gli autovettori di A. Se $\delta A \in M_{n,n}$ e' la matrice delle perturbazioni indotte su A e μ un autovalore della matrice perturbata $(A + \delta A)$, allora esiste un autovalore λ di A tale che*
>
> $$|\lambda - \mu| \leq \frac{1}{|y^T \cdot x|}\ ||\delta A||_2$$
>
> *dove x e' una colonna di X e y^T e' la corrispondente riga di X^{-1}*

Questo teorema ci dice che in sostanza la variazione dell'autovalore λ dovuta alla perturbazione δA e' proporzionale a $||\delta A||$ e che il condizionamento dipende esclusivamente dall'autovettore associato all'autovalore. Tanto piu' y^T tende ad essere *ortogonale* a x, tanto piu' l'errore tende a crescere.

9.2 Metodi iterativi

Dal momento che gli autovalori di una matrice sono gli zeri del suo polinomio caratteristico, una possibile soluzione numerica al problema degli autovalori potrebbe consistere nel trovare le radici di tale polinomio. In generale, pero', una procedura del genere non e' adeguata poiche':

- Il calcolo degli autovalori della matrice potrebbe essere ben condizionato ma non il calcolo degli zeri del suo polinomio caratteristico
- Gli errori di arrotondamento nel calcolo dei coefficienti di $P_n(\lambda)$ possono indurre elevate variazioni degli zeri del polinomio

Studieremo quindi dei metodi iterativi per il calcolo degli autovalori. Un metodo iterativo e' un metodo numerico che consiste nel costruire una successione di approssimazioni $y_1, y_2, \ldots, y_n$, che sotto opportune condizioni converge alla soluzione $\bar{x}$. Un metodo iterativo si dice *semplice* se ogni approssimazione della successione dipende dall'approssimazione costruita al passo precedente. Per un metodo iterativo valgono le definizioni di *ordine di convergenza* viste al capitolo 3. In generale valgono anche le stesse considerazioni fatte per il criterio di arresto che puo' essere

- Una tolleranza $toll$, quindi il criterio di arresto sara' $||y_k - y_{k+1}||_\infty \leq toll$.
- Un numero massimo di iterazioni $maxiter$, quindi il criterio di arresto sara' $k >= maxiter$.

Un metodo iterativo molto conosciuto per il calolo degli autovalori e' proprio il **metodo delle potenze**.

9.3 Metodo delle potenze

Il metodo delle potenze e' un metodo iterativo che permette di calcolare l'autovalore dominante (quello il modulo massimo) e un suo autovettore corrispondente. E' possibile poi modificare il metodo per poter calcolare l'autovettore con modulo minimo. Anche se il metodo calcola un solo autovalore, risulta utile per alcuni problemi quali

- Determinare l'invertibilita' di una matrice calcolando l'autovalore di minimo modulo (se dierso da 0 sara' sicuramente ivertibile)$
- Calcolo del raggio spettrale calcolando l'autovalore di massimo modulo
- In caso la matrice sia simmetrica, ottenere il numero di condizionamento in norma 2 (poiche' $||A||_2 = \sqrt{\rho(A)}$)

Inoltre, il metodo e' alla base dell'algoritmo **PageRank** di Google.

Consideriamo quindi una matrice $A_{n,n}$ che sia diagonalizzabile e che abbia quindi n autovettori $x_1, \dots, x_n$ linearmente indipendenti e autovalori $\lambda_1, \dots, \lambda_n$. Ordiniamo gli autovalori in modo decrescente nel modo seguente

$$|\lambda_1| > |\lambda_2| \geq \cdots \geq |\lambda_n|$$

Ossia (notare il primo $>$) l'autovalore con modulo massimo e' λ_1 e ha molteplicita' algebrica pari a 1. Inoltre non esistono altri autovalori con lo stesso modulo.
Fissando un vettore $t_0 \in \mathbb{R}^n$ iniziale (detto **vettore d'innesto**), sotto l'ipotesi precedente si puo' definire la successione di vettori $\{y_k\}_{k=1,2,\dots}$ nel modo seguente

$$\begin{cases} y_1 = A \cdot t_0, \\ y_k = A \cdot y_{k-1}, \quad k = 2, 3, \dots \end{cases} \quad \leftrightarrow \quad \begin{cases} y_1 = A \cdot t_0, \\ y_k = A^k \cdot t_0, \quad k = 2, 3, \dots \end{cases}$$

Spieghiamo ora come sia possibile approssimare l'autovettore λ_1 e il suo corrispondente autovettore x_1 mediante la successione appena descritta.
Dal momento che gli autovettori sono tutti linearmente indipendenti, significa che formano una base ed e' quindi possibile esprimere t_0 come combinazione lineare degli stessi, cioe'

$$t_0 = \sum_{i=1}^{n} \alpha_i x_i$$

Supponiamo inoltre che il vettore d'innesto t_0 scelto, abbia la prima componente non nulla, percio' $\alpha_1 \neq 0$. Riscriviamo t_0 come combinazione lineare all'interno della definizione del metodo, ottenendo

$$y_k = A^k \cdot t_0 = A^k \cdot (\sum_{i=1}^{n} \alpha_i x_i) = \sum_{i=1}^{n} \alpha_i A^k \cdot x_i$$

Per la seguente proprieta' degli autovalori $A^k \cdot x_i = \lambda^k x_i$, possiamo riscrivere la relazione e ottenere

$$= \sum_{i=1}^{n} \alpha_i \lambda^k \cdot x_i$$

Infine, dal momento che si vuole approssimare λ_1, per la proprieta' della linearita' mettiamo in evidenza λ_1^k e otteniamo infine

$$= \lambda_1^k \left[\alpha_1 x_1 + \sum_{i=2}^{n} \alpha_i \left(\frac{\lambda_i}{\lambda_1} \right)^k x_i \right].$$

Analogamente, otteniamo

$$y_{k+1} = \lambda_1^{k+1} \left[\alpha_1 x_1 + \sum_{i=2}^{n} \alpha_i \left(\frac{\lambda_i}{\lambda_1} \right)^{k+1} x_i \right].$$

Avendo y_{k+1} e y_k, possiamo valutare il seguente rapporto, dove $(y_k)_r$ e' l'r-*esima* componente del vettore y_k, dal momento che siccome vale per l'intero vettore,

vale anche per ogni sua componente (lo stesso vale per y_{k+1})

$$\frac{(y_{k+1})_r}{(y_k)_r} = \lambda_1 \frac{\alpha_1(x_1)_r + \sum_{i=2}^{n} \alpha_i \left(\frac{\lambda_i}{\lambda_1}\right)^{k+1} (x_i)_r}{\alpha_1(x_1)_r + \sum_{i=2}^{n} \alpha_i \left(\frac{\lambda_i}{\lambda_1}\right)^{k+1} (x_i)_r}$$

Siccome abbiamo supposto che per ipotesi λ_1 e' l'autovalore di modulo massimo, allora $\left|\frac{\lambda_i}{\lambda_1}\right| < 1,\ i = 2, \ldots, n$, per cui si puo' concludere che

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

Cioe' che da un certo indice k in poi, l'autovalore λ_1 di massimo modulo puo' essere approssimato mediante il rapporto $\frac{(y_{k+1})_r}{(y_k)_r}$.

Analogamente, possiamo ottenere l'autovettore x_1 dividendo y_k per λ_1^k e passando al limite (e' bastato eliminare λ_1^k da y_k)

$$\lim_{k\to\infty} \frac{y_k}{\lambda_1^k} = \alpha_1 x_1$$

La stessa relazione vale anche per qualsiasi componente *r-esima* come nel caso precedente

$$\lim_{k\to\infty} \frac{(y_k)_r}{\lambda_1^k} = \alpha_1(x_1)_r$$

infine, per tutti gli indici r per cui $(x_1)_r \neq 0$, si ha

$$\lim_{k\to\infty} \frac{y_k}{(y_k)_r} = \frac{x_1}{(x_1)_r}$$

Poiche' da un certo indice k in poi, l'indice m di una componente di massimo modulo di y_k rimane invariato (per le ipotesi iniziali), la successione $\frac{y_k}{(y_k)_m}$ converge all'autovettore x_1 normalizzato in norma ∞.

La convergenza delle successioni $\frac{(y_{k+1})_r}{(y_k)_r}$ e $\frac{y_k}{(y_k)_r}$ per $k \to \infty$, inoltre, dipende direttamente da quanto dista il secondo autovalore di modulo massimo λ_2 da λ_1, cioe' dal rapporto $\left|\frac{\lambda_2}{\lambda_1}\right|$. Essa sara' tanto piu' rapida tanto quando il rapporto sara' piccolo.

Si vuole ricordare inoltre che la convergenza del metodo delle potenze e' garantita se e solo se le seguenti 3 condiziioni sono verificate:

- A e' diagonalizzabile
- Il vettore d'innesto t_0 ha una componente non nulla lungo l'autovettore x_1 corrispondente all'autovalore λ_1
- L'autovalore di modulo massimo e' separato dagli altri (non ne esistono con lo stesso valore)

9.4 Metodo delle potenze con normalizzazione

Dalla dimostrazione metodo delle potenze fatta in precedenza, e' possibile osservare che per $k \to \infty$ si puo' verificare la seguente situazione

$$\begin{cases} \lambda_1^k \to 0 & \text{se } |\lambda_1| < 1 \\ \lambda_1^k \to \infty & \text{se } |\lambda_1| > 1 \end{cases}$$

Dunque se si implementa un algoritmo che lavora con l'aritmetica finita si potrebbero verificare dei fenomeni di **underflow** o **overflow** rispettivamente. Per poter evitare fenomeni di questo tipo si utilizza una versione modificata dell'algoritmo che applica ad ogni passo una normalizzazione del vettore. Vediamo inizialmente la normalizzazione a norma ∞ e successivamente quella in norma 2.

9.4.1 Normalizzazione con la norma ∞

Consideriamo sempre un vettore t_0, questa volta scelto in modo tale che $||t_0||_\infty = 1$. Costruiamo quindi la successione come nel caso precedente:

$$\begin{cases} u_k = A \cdot t_{k-1}, \\ t_k = \frac{u_k}{||u_k||_\infty}, \quad k = 1, 2, \dots \end{cases}$$

Procedendo in modo simile a prima e denotando con m l'indice di massimo modulo di u_k, cioe' $||u_k||_\infty = |(u_k)_m|$, da un certo indice k in poi la successione

$$\beta_k = \frac{(u_{k+1})_m}{(t_k)_m}$$

tende a λ_1 come $\left|\frac{\lambda_2}{\lambda_1}\right|^k$, per cui la convergenza e' piu' veloce rispetto al metodo senza normalizzazione. D'altra parte, abbiamo anche la convergenza della successione per l'autovettore

$$\lim_{k \to \infty} t_k \frac{x_1}{(x_1)_m}$$

essendo $(t_k)_m \pm 1$, abbiamo che la successione converge all'autovettore x_1 normalizzato in norma ∞.

9.4.2 Normalizzazione con la norma 2

Anche in questo caso, consideriamo sempre un vettore t_0, sempre scelto in modo tale che $||t_0||_2 = 1$, e consideriamo la successione

$$\begin{cases} u_k = A \cdot t_{k-1}, \\ t_k = \frac{u_k}{||u_k||_2}, \quad k = 0, 1, 2, \dots \end{cases}$$

In questo caso si puo' usare come approssimazione dell'autovalore λ_1 il quoziente di Rayleigh

$$\sigma_k = \frac{t_k^T A t_k}{t_k^T t_k} = t_k^T u_{k+1}.$$

Dunque si ottiene

$$\lim_{k\to\infty} \sigma_k = \lambda_1 \quad \text{e} \quad \lim_{k\to\infty} t_k = \frac{x_1}{||x_1||_2}$$

e anche in questo caso la convergenza dipende dal rapporto $\left|\frac{\lambda_2}{\lambda_1}\right|^k$.

La normalizzazione in norma 2, pero', risulta piu' conveniente in casi in cui la matrice A e' normale. E' dimostrabile che in questi casi, $\sigma_k \to \lambda_1$ con $k \to \infty$ come $\left|\frac{\lambda_2}{\lambda_1}\right|^{2k}$.

Siccome il metodo delle potenze richiede ad ogni passo la moltiplicazione di una matrice per un vettore, il suo costo computazionale e' di $\approx kn^2$ flops. Il metodo inoltre puo' essere modificato sia per calcolare l'autovalore minimo, sia per migliorare delle approssimazioni degli autovalori che sono state localizzate mediante teoremi di localizzazione quali Gerschgorin o Hirsch. Vediamo ora piu' nel dettaglio in cosa consistono tali modifiche.

9.5 Metodo delle potenze inverse

Come detto in precedenza, il metodo delle potenze puo' essere modificato per permettere di calcolare l'autovalore di modulo minimo. L'idea alla base della modifica e' che se A e' una matrice che ha autovalori

$$|\lambda_1| \geq |\lambda_2| \geq \cdots \geq |\lambda_{n-1}| > |\lambda_n| > 0$$

Ossia che λ_n e' l'autovalore di minimo modulo con molteplicita' algebrica pari a 1, e che non esistono autovalori con lo stesso modulo. Per la proprieta' degli autovalori, la matrice inversa A^{-1} avra' i seguenti autovalori

$$\left|\frac{1}{\lambda_n}\right| > \left|\frac{1}{\lambda_{n-1}}\right| \geq \cdots \geq \left|\frac{1}{\lambda_2}\right| \geq \left|\frac{1}{\lambda_1}\right|$$

per cui per approssimare l'autovalore di minimo modulo di A, e' sufficiente calcolare l'autovalore di massimo modulo di A^{-1} e farne l'inverso.
In sostanza il metodo sara' analogo al metodo delle potenze normali, con la differenza che si avra' A^{-1} al posto di A, per cui la successione (normalizzata in norma ∞) risulta essere la seguente:

$$\begin{cases} u_k = A^{-1} \cdot t_{k-1}, \\ t_k = \frac{u_k}{||u_k||_\infty}, \quad k = 1, 2, \ldots \end{cases}$$

Per calcolare quindi λ_n si considera l'inverso della successione β_k, poiche' in caso contrario si otterrebbe $\frac{1}{\lambda_n}$

$$\frac{(t_k)_m}{(u_{k+1})_m}$$

Il problema del metodo e' che richiede di conoscere l'inversa di A. Nell'implementazione dell'algoritmo non si calcola direttamente la matrice inversa, perche' potrebbe essere troppo costoso, ma si risolve ad ogni passo

il sistema lineare (basta portare A^{-1} al primo membro nella prima equazione della successione)

$$Au_k = t_{k-1}$$

nell'incognira u_k. Il costo computazionale diventa cosi' $\approx k\frac{n^3}{3}$, mentre con la fattorizzazione LU prima di eseguire l'algoritmo e risolvendo cosi' due sistemi triangolari ad ogni ciclo e' $\frac{n^3}{3} + kn^2$.

9.6 Metodo delle potenze inverse con shift

Il metodo delle potenze inverse con shift ci permette di migliorare un'approssimazione iniziale di un autovalore ad ogni passo. Tale approssimazione potrebbe essere stata localizzata mediante i teoremi visti in precedenza, in modo da rendere la convergenza del metodo piu' veloce.

Osserviamo innanzitutto, che se (λ, x) e' autocoppia (cioe' autovalore e autovettore associati) di A, allora $(\lambda - p, x)$ e' autocoppia di $(A - pI)$:

$$Ax = \lambda x \quad \rightarrow \quad (A - pI)x = (\lambda - p)x \quad \rightarrow \quad (A - pI)^{-1}x = (\lambda - p)^{-1}x$$

Quindi, se p e' un'approssimazione (molto scadente) di λ, possiamo ottenere un'approssimazione con cifre piu' significative. Infatti, quanto piu' p si avvicina a λ, tanto piu' il modulo del rapporto

$$\mu = \left| \frac{1}{\lambda - p} \right|$$

diventa grande. Quindi, calcolando l'autovalore μ di modulo massimo della matrice $(A - pI)^{-1}$ usando il metodo delle potenze con

$$u_k = (A - pI)^{-1}t_{k-1} \quad \rightarrow \quad (A - pI)u_k = t_{k-1}$$

Quindi sempre risolvendo il sistema lineare con $(A - pI)$ matrice dei coefficienti come nel caso precedente. Si nota fin da subito quindi che

$$p + \frac{1}{\mu} \quad \rightarrow \quad p + (\lambda - p) \quad \rightarrow \quad \lambda$$

fornisce un'approssimazione piu' accurata di λ.

10 Metodi iterativi per la soluzione di sistemi lineari

10.1 Introduzione

Nel capitolo 7 abbiamo visto i metodi diretti per la soluzione di sistemi lineari. Come gia' detto, pero', questa tipologia di metodi non e' particolarmente indicata per sistemi lineari con matrici dei coefficienti *sparse*, dal momento che soffrono principalmente del cosiddetti problema di *fill-in* (e una conseguente saturazione della memoria). I metodi iterativi, lasciano invece la matrice dei coefficienti invariata, evitando cosi' il fill-in in casi di matrici sparse. Come detto gia' in precedenza, un metodo iterativo consiste nel generare una successione di vettori $x_k, k \geq 0$ tale che *converga* alla soluzione esatta x del sistema $Ax = b$.

Un approccio iniziale per derivare un metodo iterativo potrebbe essere analogo a quello adottato nel metodo di punto fisso. Il ragionamento lo stesso, ma si applica a sistemi lineari. Sia quindi A una matrice dei coefficienti non singolare, allora possiamo fare il seguente ragionamento

$$Ax = b \quad \leftrightarrow \quad b - Ax = 0 \quad \leftrightarrow \quad \underbrace{x + b - Ax}_{\psi(x)} = x$$

per un'opportuna funzione di iterazione $\psi(x) = (I - A)x + b$ (raccolti x da A e x). Nel metodo di punto fisso, la convergenza del metodo era garantita se la derivata prima della funzione di iterazione in valore assoluto era minore di 1. In questo caso, l'analogo della derivata e' il raggio spettrale, per cui per la convergenza deve valere che $\rho(I - A) < 1$. Possiamo quindi definire la successione degli x_k nel modo seguente (porto a destra x e sommo x in entrambe le parti)

$$\begin{cases} x^{(0)} \ \rightarrow \ \textit{approssimazione iniziale} \\ x^{(k+1)} = \psi(x^{(k)}) \rightarrow x^{(k)} + \underbrace{b - Ax^{(k)}}_{r^{(k)}} \end{cases}$$

Abbiamo quindi ottenuto un metodo di punto fisso per sistemi lineari, in cui $r^{(k)}$ e' il residuo della soluzione al passo *k-esimo*. E' possibile pero' precondizionare ulteriormente il metodo, cioe' aggiungere delle accortezze per rendere piu' veloce la sua convergenza.

Sia P una matrice invertibile "semplice", cioe' che sia triangolare (s/i) oppure diagonale. Per il momento ci limiteremo a dare solamente queste informazioni riguardo a P. E' possibile quindi moltiplicare ambo i membri del sistema per P^{-1}, ottenendo

$$P^{-1}Ax = P^{-1}b \ \rightarrow \ 0 = P^{-1}b - P^{-1}Ax \ \rightarrow \ x = \underbrace{x + P^{-1}b - P^{-1}Ax}_{\psi(x)}$$

Notiamo che questa volta possiamo ottenere due funzioni $\psi(x)$ di iterazione

differenti, a seconda che si voglia raccogliere P^{-1} oppure x

$$\psi_1(x) = (I - P^{-1}A)x + P^{-1}b$$
$$\psi_2(x) = x + P^{-1}(b - Ax)$$

Scegliamo la seconda, e come prima costruiamo la successione risultante

$$\begin{cases} x^{(0)} \ \rightarrow \ \textit{approssimazione iniziale} \\ x^{(k+1)} = \psi(x^{(k)}) \rightarrow x^{(k)} + P^{-1}(\underbrace{b - Ax^{(k)}}_{r^{(k)}}) \\ \end{cases}$$

Ma quindi, la seconda equazione puo' essere scritta come

$$x^{(k+1)} = x^{(k)} + P^{-1}r^{(k)}$$

Siccome il calcolo della matrice inversa e' molto oneroso in termini computazionali ed e' generalmente instabile numericamente, e' possibile evitare di calcolarla riscrivendo la relazione come sistema lineare

$$x^{(k+1)} = x^{(k)} + P^{-1}r^{(k)} \ \rightarrow \ Px^{(k+1)} = Px^{(k)} + r^{(k)} \ \rightarrow \ P(\underbrace{x^{(k+1)} - x^{(k)}}_{y}) = r^{(k)}$$

Si ottiene quindi un sistema lineare con matrice dei coefficienti P, per cui calcolarne la soluzione e' facile dal momento che l'ipotesi e' che sia triangolare. Una volta risolto il sistema e trovata la soluzione y e' possibile ottenere $x^{(k+1)}$, poiche'

$$x^{(k+1)} = x^{(k)} + y$$

Per cui e' possibile costruire il metodo iterativo seguente

1. $r^{(k)} = b - Ax^{(k)}$
2. Risolvo il sistema lineare $Py = r^{(k)}$
3. Calcolo la $k+1$-*esima* soluzione $x^{(k+1)} = y + x^{(k)}$

con $k = 1, 2, \ldots$

10.1.1 Convergenza

Studiamo ora la convergenza di tale metodo ottenuto. Per definizione, l'errore al passo k-*esimo* e' $e^{(k)} = x^{(k)} - x$, quindi

$$\begin{aligned}
e^{(k+1)} &= x^{(k+1)} - x && (\textit{definizione errore}) \\
&= x^{(k)} + P^{-1}(b - Ax^{(k)}) - x && (\textit{definizione } x^{(k+1)}) \\
&= x^{(k)} - x + P^{-1}A(x - x^{(k)}) && (\textit{esplicito } A) \\
&= I(x^{(k)} - x) - P^{-1}A(x - x^{(k)}) && (\textit{esplicito } I) \\
&= (I - P^{-1}A)(x^{(k)} - x) && (\textit{esplicito } (x^{(k)} - x)) \\
&= (I - P^{-1}A)e^{(k)} && (\textit{definizione errore})
\end{aligned}$$

Procedendo nello stesso modo, possiamo ri-espandere $e^{(k)}$ iterativamente, ottenendo in definitiva

$$= (I - P^{-1}A)^{k+1}e^{(0)}$$

E' possibile dimostrare che se $\rho(I - P^{-1}A) < 1$, allora si ha che $e^{(k)} \to 0$ per $k \to \infty$. La condizione seguente e' la condizione necessaria e sufficiente per la convergenza del metodo. Una condizione di piu' facile verifica ma piu' *"debole"* (e quindi solo sufficiente) e' che la convergenza si ha se $||I - P^{-1}A||_{1,\infty} < 1$.

Stabilite le condizioni di convergenza, vediamo ora come scegliere la matrice di condizionamento P scelta in precedenza. In generale, tale matrice dovra' esser scelta in modo tale che:

- $\rho(I - P^{-1}A)$ deve essere il piu' piccolo possibile
- $Py = r^{(k)}$ deve essere un sistema molto semplice da risolvere

Osserviamo pero' che le due scelte sono in qualche modo contrastanti. Supponiamo di scegliere $P = A$, in quanto ne risulterebbe un raggio spettrale piu' piccolo, per cui $\rho(I - A^{-1}A) \to \rho(I - I) \to \rho(0) = 0$. Il problema e' che tale scelta porta alla soluzione del sistema $Py = Ay$, per cui si sarebbe di nuovo da capo. Questo caso limite, pero', ci da un indizio sulla direzione da prendere per la scelta di P: deve essere "simile" ad A ma piu' *"semplice"*.

10.2 Metodo di Jacobi

Il metodo di Jacobi e' un metodo iterativo in cui la matrice di preceondizionamento P e' la matrice formata dalla diagonale principale di A.

$$P = \begin{bmatrix} a_{11} & 0 & 0 & 0 \\ 0 & a_{22} & 0 & 0 \\ 0 & 0 & \ddots & 0 \\ 0 & 0 & 0 & a_{nn} \end{bmatrix}$$

Dal momento che la matrice e' non singolare per ipotesi, e' sempre possibile ottenere la sua matrice diagonale scambiando le righe che diversamente condurebbero ad avere un elemento nullo sulla diagonale principale. Se questo non fosse vero ci sarebbe qualche riga o qualche colonna interamente composta da 0, e questo rendrebbe A singolare e quindi di conseguenza il sistema non sarebbe risolvibile.

Consideriamo ora la funzione di iterazione $\psi(x) = (I - D^{-1}A)x^{(k)} + D^{-1}b$ e verifichiamo l'ipotesi di convergenza:

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & 1 \end{bmatrix} - \begin{bmatrix} 1/a_{11} & 0 & 0 & 0 \\ 0 & 1/a_{22} & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & 1/a_{nn} \end{bmatrix} \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} \end{bmatrix}$$

Ricordando che moltiplicare la matrice diagonale per un'altra matrice a sinistra equivale a moltiplicare l'elemento della diagonale per le righe, e sottraendo infine il risultato per la matrice identita', si ottiene la matrice seguente

$$\begin{bmatrix} 0 & -a_{12}/a_{11} & \cdots & -a_{1n}/a_{11} \\ -a_{21}/a_{22} & 0 & \cdots & -a_{1n}/a_{22} \\ \vdots & \vdots & \ddots & \vdots \\ -a_{n1}/a_{nn} & -a_{n2}/a_{nn} & \cdots & 0 \end{bmatrix}$$

Avendo quindi la forma della matrice di iterazione nota, possiamo procedere a vedere se il metodo possa convergere o meno:

$$||I - D^{-1}A||_\infty \leftrightarrow \max_{1\leq i\leq n} \sum_{j=1, j\neq i}^{n} \left|\frac{a_{ij}}{a_{ii}}\right| < 1 \qquad (\textit{definizione norma } \infty)$$

$$\leftrightarrow |a_{ii}| > \sum_{j=1, j\neq i}^{n} |a_{ij}| \quad i = 1, \ldots, n \qquad (\textit{esplicito } a_{ii})$$

Ma questo succede se e solo se la matrice e' diagonalmente dominante stretta per righe, per cui possiamo concludere che il metodo convergera' per tutte le matrici dei coefficienti diagonalmente dominanti strette.

Una volta stabilito che la scelta di $P = D$ e' sensata (dal momento che e' convergente), consideriamo ora la successione

$$\begin{cases} x^{(0)} \rightarrow \textit{approssimazione iniziale} \\ x^{(k+1)} = (I - D^{-1}A)x^{(k)} + D^{-1}b \end{cases}$$

estendendo la somma e il prodotto tra matrici/vettori otteniamo che il generico elemento i del vettore $x^{(k+1)}$ e' dato dalla formula

$$x_i^{(k+1)} = -\sum_{j=1, j\neq i}^{n} \frac{a_{ij}}{a_{ii}} x_j^{(k)} + \frac{1}{a_{ii}} b_i$$

Possiamo osservare che ad ogni iterazione il metodo dovra' mantenere in memoria due variabili: il vettore $x^{(k)}$ e il vettore $x^{(k+1)}$.

10.3 Metodo di Gauss-Seidel

E' una modifica del metodo di Jacobi che utilizza un solo vettore ad ogni iterazione (e non due). L'idea e' quella di sfruttare le componenti $k+1$-*esime* che sono gia' state calcolate. In altri termini

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

Dall'espressione si nota come sia molto simile a quella del metodo di Jacobi, con la differenza che si sfruttano le componenti ($k+1$-*esime*)precedenti all'iterazione *i-esima* (prima sommatoria).

$$a_{ii}x_i^{(k+1)} + \sum_{j=1}^{i-1} a_{ij}x_j^{(k+1)} = b_i - \sum_{j=i+1}^{n} a_{ij}x_j^{(k)} \qquad (\textit{moltiplico per } a_{ii})$$

$$\sum_{j=1}^{i} a_{ij}x_j^{(k+1)} = b_i - \sum_{j=i+1}^{n} a_{ij}x_j^{(k)} \qquad (\textit{linearita' sommatoria})$$

L'ultima relazione ottenuta in forma matriciale corrisponde al sistema lineare seguente

$$\underbrace{\begin{bmatrix} a_{11} & 0 & \dots & 0 \\ a_{21} & a_{22} & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n1} & \dots & a_{nn} \end{bmatrix}}_{L} x^{(k+1)} = b - \underbrace{\begin{bmatrix} 0 & a_{12} & a_{13} & \dots & a_{1n} \\ 0 & 0 & a_{23} & \dots & a_{2n} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \dots & a_{nn} \end{bmatrix}}_{A-L} x^{(k)}$$

Il che equivale in forma compatta a

$$Lx^{(k+1)} = b - (A - L)x^{(k)}$$

Per ottenere la funzione di iterazione ψ, manipoliamo algebricamente quest'ultima relazione

$$\begin{aligned} Lx^{(k+1)} &= b - (A - L)x^{(k)} \\ &= L^{-1}(L - A)x^{(k)} + L^{-1}b \qquad (\textit{moltiplico per } L^{-1}) \\ &= (I - L^{-1}A)x^{(k)} + L^{-1}b \qquad (\textit{espando la prima moltiplicazione}) \end{aligned}$$

Abbiamo quindi trovato la relazione di ψ, da cui si nota subito la similarita' rispetto al metodo di Jacobi. Piu' precisamente, il metodo di Gauss-Seidel differisce dal metodo di Jacobi solo nella scelta della matrice di precondizionamento, dal momento che in Jacobi e' D, mentre in GS e' L.
Si puo' dimostrare che il metodo di Gauss-Seidel converge:

- Se e solo se $\rho(I - L^{-1}A) < 1$
- Se A e' strettamente diagonalmente dominante
- Se A e' simmetrica e definita positiva (se tutti i suoi autovalori sono definiti positivi)

Inoltre, e' importante sottolineare il fatto che la convergenza di Gauss-Seidel e quella di Jacobi non sono in qualche modo correlazionate. Cio' implica che la convergenza di un metodo non assicura la convergenza anche dell'altro e viceversa. Generalmente, pero', quando entrambi convergono Gauss-Seidel ha una convergenza superiore.

Un caso specifico in cui questo non accade e' per matrici tri-diagonali, cioe' matrici in cui gli elementi sulle 3 diagonali sono diversi da 0, mentre tutti gli altri elementi sono nulli. In questo caso e' stato dimostrato che entrambi i metodi sono convergenti o divergenti, e in caso ci sia la convergenza vale

$$\rho(I - L^{-1}A) = (\rho(I - D^{-1}A))^2$$

Cio' significa che asintoticamente sono necessarie meta' iterazioni del metodo di Gauss-Seidel per ottenere la stessa precisione del metodo di Jacobi. In altri termini, la velocita' di convergenza del metodo Gauss-Seidel e' doppia rispetto a quella di Jacobi.

10.4 Criterio d'arresto

Come ogni metodo iterativo, Gauss-Seidel e Jacobi dovranno avere un qualche criterio d'arresto che puo' essere dato da una tolleranza e da un numero massimo di iterazioni. Nel caso della tolleranza, l'idea e' sempre quella di fermarsi quando l'errore della soluzione $x^{(k)}$ e' minore di una certa tolleranza τ. Dal momento che l'errore e' calcolato in termini della soluzione esatta, sono necessari degli stimatori *"a posteriori"* basati sul:

1. Residuo al passo *k-esimo* $r^{(k)}$
   - Errore assoluto:
   $$||x^{(k)} - x|| \leq ||A^{-1}||\,||r^{(k)}||$$
   - Errore relativo:
   $$\frac{||x^{(k)} - x||}{||x||} \leq \kappa(A)\varepsilon$$
2. Incremento $\delta^{(k)} = x^{(k+1)} - x^{(x)}$
   - Errore assoluto:
   $$||e^{(k)}||_2 \leq \frac{||\delta^{(k)}||_2}{1 - \rho(B)} \leq \frac{\varepsilon}{1 - \rho(B)}$$
   - Errore relativo:
   $$\frac{||e^{(k)}||}{||b||} \leq \frac{\varepsilon}{1 - \rho(B)}$$

Nel secondo caso il controllo dell'incremento e' significativo soltanto se $\rho(B)$ e' molto piu' piccolo di 1, poiche' in tal caso l'errore sara' dello stesso ordine di grandezza dell'incremento.