

Approssimazione di dati e funzioni

Sandra Pieraccini

Politecnico di Torino, Dipartimento di Scienze Matematiche
sandra.pieraccini@polito.it
<http://calvino.polito.it/~pieraccini>

Calcolo Numerico e MATLAB

Ultimo aggiornamento: 21 novembre 2016



Argomenti trattati

- 1 Interpolazione polinomiale
 - Base di Lagrange
 - Condizionamento
 - Convergenza
 - Base di Newton
- 2 Interpolazione polinomiale a tratti
- 3 Metodo dei minimi quadrati



Generalità

Problema 1 Dati (x_i, y_i) $i = 0, \dots, n$ (es. misurazioni) voglio approssimarli con una funzione g in modo da poter stimare l'andamento dei dati anche in punti $x \neq x_i$.

Esempio:

Le centralina posta presso il Lingotto ha rilevato per il 13/11/2015 le seguenti concentrazioni di Ossido di Azoto nell'aria (microgrammi/metro cubo), a intervalli di un'ora (fonte: *Banca Dati regionale della qualità dell'aria* della Regione Piemonte):

Ora	1	2	3	4	5	6	7	8	9	10
NO	243	209	181	179	180	166	163	157	187	192
Ora	11	12	13	14	15	16	17	18	19	20
NO	138	95	56	32	21	12	11	61	146	186

Si vuole stimare il livello di concentrazione di NO alle ore 14:15 e 20:30.

Problema 2 Data f voglio approssimarla con una g più “facile” da usare (ad esempio un polinomio).

Esempio:

$$\int_a^b \exp(-x^2) dx = ?$$

Se $\exp(-x^2) \simeq g(x)$ su $[a, b]$, e g so integrarla facilmente, posso pensare di approssimare

$$\int_a^b \exp(-x^2) dx \simeq \int_a^b g(x) dx$$

NB Problema 2 si tratta come il Problema 1 con dati $(x_i, f(x_i))$, $i = 0, \dots, n$



Data $f \in \mathbb{F} = C^k[a, b]$, per approssimarla occorrono due ingredienti:

- 1 Individuare un **sottospazio** \mathbb{F}_m di funzioni di dimensione finita in cui scegliere la funzione f_m che approssima f

- 1 \mathbb{P}_m , polinomi algebrici di grado m :

$$f_m(x) = p_m(x) = \sum_{k=0}^m a_k x^k \quad (\dim(\mathbb{P}_m) = m + 1)$$

- 2 \mathbb{P}_m^T , funzioni polinomiali a tratti, costituite da N tratti polinomiali di grado r (tipicamente basso): $f_m(x)|_{[x_i, x_{i+1}]} \in \mathbb{P}_r$
($\dim(\mathbb{P}_m^T) = (r + 1) \times N$)

- 3 \mathbb{S}_r , funzioni spline di ordine r (caso particolare di 2)

- 4 Π_m , polinomi trigonometrici:

$$\pi_m(x) = a_0 + \sum_{k=1}^m (a_k \cos(k\omega x) + b_k \sin(k\omega x))$$

$$(\dim(\Pi_m) = 2m + 1)$$

- 5 ...



- ③ Individuare un criterio per determinare $f_m \in \mathbb{F}_m$
 - ① interpolazione: $f_m(x_i) = y_i, i = 0, \dots, n$
 - ② minimi quadrati: $\min \sum_{i=0}^n (f_m(x_i) - y_i)^2$ (ideale per approssimazione di dati sperimentali)



Osservazione

*Dimensione = # elementi della base di \mathbb{F}_m
= # parametri che individua $f_m \in \mathbb{F}_m$
= # di condizioni da imporre*

*Nel caso dell'interpolazione, le condizioni di interpolazione sono tante quante i nodi ($n + 1$), e il grado **massimo** del polinomio che si può costruire è quindi n , poiché \mathbb{P}_n è uno spazio vettoriale di dimensione $n + 1$.*



Misura dell'errore

Norma di funzione:

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

Facendo crescere m mi aspetto che l'approssimazione migliori.

Definizione (convergenza)

Dati $\{\mathbb{F}_m\}_{m \geq 0}$, e $f_m \in \mathbb{F}_m$, se

$$\lim_{m \rightarrow \infty} \|f - f_m\|_{\infty} = 0$$

si dice che si ha convergenza (uniforme) di f_m a f (i.e. convergenza dell'approssimazione)



- Sono assegnati i dati (x_i, y_i) , $i = 0, \dots, n$; x_i sono detti **nod**.
- Vogliamo costruire il polinomio $p_n(x) \in \mathbb{P}_n$ che interpola i dati, quindi tale che $p_n(x_i) = y_i$ per $i = 0, \dots, n$
- Scriviamo $p_n(x)$ nella forma

$$p_n(x) = \sum_{k=0}^n c_k x^k$$

- Questo corrisponde a considerare la base monomiale

$$\mathbb{P}_n = \text{span}\{1, x, x^2, \dots, x^n\}$$

Attenzione!

La base monomiale è solo **una possibile** base. Più in generale,

$$\mathbb{P}_n = \text{span}\{\phi_0(x), \dots, \phi_n(x)\}$$

dove $\phi_i(x)$ sono $n + 1$ polinomi linearmente indipendenti.

Imponiamo il criterio di interpolazione per determinare il polinomio: per $i = 0, \dots, n$

$$p_n(x_i) = y_i \Rightarrow \sum_{k=0}^n c_k x_i^k = y_i$$

Posto:

$$A = \begin{pmatrix} 1 & x_0 & \dots & x_0^n \\ 1 & x_1 & \dots & x_1^n \\ \vdots & \vdots & \ddots & \vdots \\ 1 & x_n & \dots & x_n^n \end{pmatrix} \in \mathbb{R}^{(n+1) \times (n+1)}$$

$$c = (c_0, \dots, c_n)^T \in \mathbb{R}^{n+1}, \quad y = (y_0, \dots, y_n)^T \in \mathbb{R}^{n+1}$$

$$\implies Ac = y$$



Esistenza e unicità polinomio interpolante \Leftrightarrow invertibilità di A

Proprietà

A : matrice di Vandermonde (comando MATLAB `vander`).

Se i nodi x_i sono distinti, si può dimostrare che A è invertibile.

Corollario

Se i nodi sono distinti, il polinomio interpolante esiste ed è unico, e in linea di principio i coefficienti si possono determinare risolvendo il sistema lineare $Ac = y$. Ma...

Esercizio proposto

Si provi a costruire il polinomio interpolante i dati sull'esempio dell'ossido di azoto. Si analizzi numericamente la matrice A ottenuta.



Come trovo quindi a_k ? **NON** risolvo il sistema lineare:

- 1 può essere molto mal condizionato (es. A = matrice di Vandermonde con base monomiale)
- 2 non conviene computazionalmente

Basi diverse da quella monomiale \implies metodi diversi per determinare coefficienti della combinazione lineare



Quanto visto in precedenza si può generalizzare a basi diverse:
considero

$$\mathbb{P}_n = \text{span}\{\phi_0(x), \phi_1(x), \dots, \phi_n(x)\}$$

$$p_n(x) = \sum_{k=0}^n c_k \phi_k(x)$$

$$p_n(x_i) = y_i, \quad i = 0, \dots, n \Rightarrow \sum_{k=0}^n c_k \phi_k(x_i) = y_i$$

$$A = \begin{pmatrix} \phi_0(x_0) & \phi_1(x_0) & \dots & \phi_n(x_0) \\ \phi_0(x_1) & \phi_1(x_1) & \dots & \phi_n(x_1) \\ \vdots & \vdots & \ddots & \vdots \\ \phi_0(x_n) & \phi_1(x_n) & \dots & \phi_n(x_n) \end{pmatrix} \in \mathbb{R}^{(n+1) \times (n+1)}$$
$$\Rightarrow Ac = y$$



Proposizione

Se $\phi_k(x)$, $k = 0, \dots, n$ linearmente indipendenti e $x_i \neq x_j$ per $i \neq j$ (nodi distinti) la matrice A è invertibile.

Attenzione!

Cambiare la base, non cambia il polinomio interpolante (che è unico!), cambia solo la sua rappresentazione.



Base di Lagrange

Consideriamo polinomi $\ell_k(x)$, $k = 0, \dots, n$, caratterizzati da:

$$\ell_k(x_i) = \delta_{ik} = \begin{cases} 1 & i = k \\ 0 & i \neq k \end{cases}$$

$$p_n(x) = \sum_{k=0}^n c_k \ell_k(x)$$

Imponiamo le condizioni di interpolazione $p_n(x_i) = y_i$, $i = 0, \dots, n$

$$\sum_{k=0}^n c_k \ell_k(x_i) = y_i$$

$$c_i \ell_i(x_i) = y_i$$

$$c_i = y_i!!!$$



Come è fatto $\ell_k(x)$?

Poiché deve azzerarsi in tutti i nodi tranne x_k , sarà

$$\ell_k(x) = \text{costante} \times \prod_{i \neq k} (x - x_i)$$

Poiché deve valere 1 nel nodo x_k , sarà

$$\ell_k(x) = \frac{\prod_{i \neq k} (x - x_i)}{\prod_{i \neq k} (x_k - x_i)}$$

Esempio:

$(x_0, y_0) = (-2, 3)$, $(x_1, y_1) = (1, -7)$, $(x_2, y_2) = (3, -5)$:

$$\ell_0(x) = \frac{(x-1)(x-3)}{(-2-1)(-2-3)}, \quad \ell_1(x) = \frac{(x+2)(x-3)}{(1+2)(1-3)}, \quad \ell_2(x) = \frac{(x+2)(x-1)}{(3+2)(3-1)}$$

$$p_2(x) = 3\ell_0(x) - 7\ell_1(x) - 5\ell_2(x)$$



Esercizio proposto

Scrivere uno script per visualizzare i polinomi di Lagrange costruiti su 5 nodi (distinti) a scelta.



- La forma di Lagrange del polinomio interpolante ci consente di studiare in modo semplice il condizionamento del problema dell'interpolazione polinomiale.
- Siano y_i , $i = 0, \dots, n$ i dati dell'interpolazione, e $\bar{y}_i = y_i + \delta_i$ dei dati perturbati.
- Sia $p_n(x)$ il polinomio che interpola i dati y_i , e $\bar{p}_n(x)$ il polinomio che interpola i dati \bar{y}_i . Distanza relativa $|p_n(x) - \bar{p}_n(x)|$ rispetto a δ_i ?



Abbiamo

$$p_n(x) = \sum_{k=0}^n y_k \ell_k(x) \quad \bar{p}_n(x) = \sum_{k=0}^n \bar{y}_k \ell_k(x)$$

$$p_n(x) - \bar{p}_n(x) = \sum_{k=0}^n (y_k - \bar{y}_k) \ell_k(x)$$

$$|p_n(x) - \bar{p}_n(x)| \leq \sum_{k=0}^n |\delta_k| |\ell_k(x)| \leq \max_{0 \leq k \leq n} |\delta_k| \sum_{k=0}^n |\ell_k(x)|$$

$$\|p_n - \bar{p}_n\|_{\infty} \leq \|\delta\|_{\infty} \left\| \sum_{k=0}^n |\ell_k(x)| \right\|_{\infty}$$

$$\|p_n - \bar{p}_n\|_{\infty} \leq \|\delta\|_{\infty} \Lambda_n$$

$\Lambda_n = \left\| \sum_{k=0}^n |\ell_k(x)| \right\|_{\infty}$: costante di Lebesgue, dipende solo dai nodi



$$\|p_n - \bar{p}_n\|_\infty \leq \|\delta\|_\infty \Lambda_n$$

Errori assoluti. Adesso, i denominatori:

$$\begin{aligned}\|p_n\|_\infty &= \max_{x \in [a,b]} |p_n(x)| \geq \max_{x \in \{x_0, \dots, x_n\}} |p_n(x)| \\ &= \max_{0 \leq k \leq n} |p_n(x_k)| = \max_{0 \leq k \leq n} |y_k| = \|y\|_\infty\end{aligned}$$

$$\frac{\|p_n - \bar{p}_n\|_\infty}{\|p_n\|_\infty} \leq \Lambda_n \frac{\|\delta\|_\infty}{\|y\|_\infty}$$

$$\Lambda_n = \left\| \sum_{k=0}^n |\ell_k(x)| \right\|_\infty$$

funge da numero di condizionamento per il problema dell'interpolazione polinomiale.



Convergenza dell'interpolazione polinomiale

Definizione

Errore di interpolazione:

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

Sicuramente:

- ① $E_n(x_i) = 0$
- ② $E_n(x) \equiv 0$ se f è un polinomio di grado $\leq n$

Ma in generale che accade a

$$\lim_{n \rightarrow \infty} \|E_n(x)\|_{\infty}?$$

Dipende dalla scelta nodi e dalle caratteristiche di f !



Teorema

Se $f \in C[a, b]$

$$\|E_n\|_\infty \leq (1 + \Lambda_n) \min_{q \in \mathbb{P}_n} \|f - q\|_\infty$$

$\min_{q \in \mathbb{P}_n} \|f - q\|_\infty$: *errore di migliore approssimazione, dipende solo da f*

Come si comportano $1 + \Lambda_n$ e $\min_{q \in \mathbb{P}_n} \|f - q\|_\infty$?



Teorema (Weierstrass)

Se $f \in C[a, b] \forall \varepsilon > 0 \exists p_n(x), n = n(\varepsilon)$ t.c.

$$|f(x) - p_n(x)| < \varepsilon \quad \forall x \in [a, b]$$

Teo Weierstrass implica $\min_{q \in \mathbb{P}_n} \|f - q\|_\infty \rightarrow 0$ per $n \rightarrow \infty$

$$\|E_n\|_\infty \leq (1 + \Lambda_n) \underbrace{\min_{q \in \mathbb{P}_n} \|f - q\|_\infty}_{\rightarrow 0}$$

Purtroppo, $\Lambda_n \rightarrow \infty$ per $n \rightarrow \infty$!

① $\Lambda_n \geq e^{\frac{n}{2}}$ con nodi equidistanti

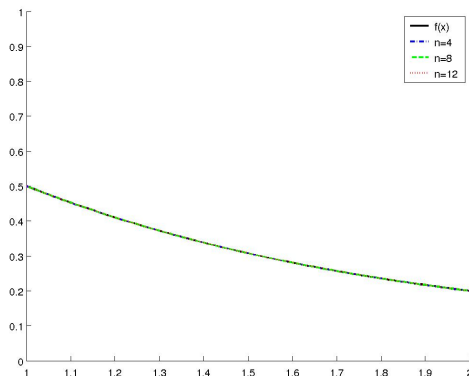
② $\Lambda_n \sim \frac{\pi}{2} \log n$ con nodi di Chebichev:

$$x_i = \cos\left(\frac{2i+1}{2(n+1)}\pi\right) \in (-1, 1), \quad i = 0, \dots, n$$



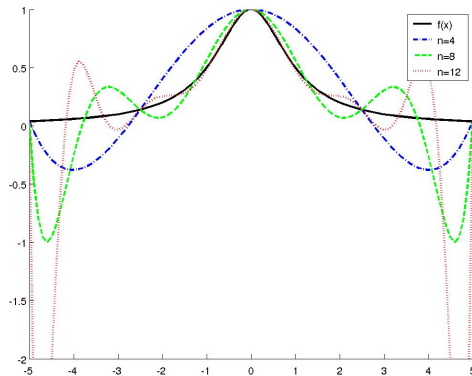
Esempio: Fenomeno di Runge

$$f(x) = \frac{1}{1+x^2}, \quad x \in [a, b]$$



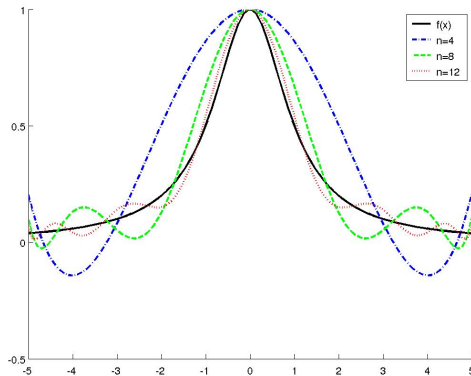
$x \in [1, 2]$, nodi equidistanti





$x \in [-5, 5]$, nodi equidistanti





$x \in [-5, 5]$, nodi Chebichev



Teorema (Bernstein)

Se $f \in C^1[a, b]$ la successione di polinomi interpolanti costruiti su nodi di Chebycev converge uniformemente a f in $[a, b]$.

Se $f \in C^2[a, b]$, si ha inoltre $\|E_n\|_\infty = \mathcal{O}(\frac{1}{\sqrt{n}})$

Esercizio proposto

Scrivere uno script per sperimentare il verificarsi del fenomeno di Runge interpolando la funzione

$$f(x) = \log(25x^2 + 1)$$

nell'intervallo $[-1, 1]$ usando i comandi `polyfit` e `polyval`



Esercizio proposto

Scrivere uno script per interpolare i dati relativi all'Ossido di Azoto usando i comandi `polyfit` e `polyval`



Base di Newton

- Sia $p_{n-1}(x)$ interpolante (x_i, y_i) , $i = 0, \dots, n-1$
- Aggiungo (x_n, y_n) e cerco $p_n(x)$ sfruttando $p_{n-1}(x)$:

$$p_n(x) = p_{n-1}(x) + q_n(x), \quad q_n(x) = ???$$

- Che sappiamo di $q_n(x)$?
 - ① è un polinomio di grado n
 - ② $p_n(x_i) = p_{n-1}(x_i) + q_n(x_i)$, $i = 0, \dots, n-1$

$$y_i = y_i + q_n(x_i) \implies q_n(x_i) = 0, \quad i = 0, \dots, n-1$$

$q_n(x)$ ha come radici x_i , $i = 0, \dots, n-1$



- Si ha

$$q_n(x) = \text{costante} \times (x - x_0)(x - x_1) \dots (x - x_{n-1}) =: \text{cost.} \times \omega_n(x)$$

$$\omega_n(x) := \overbrace{(x - x_0)(x - x_1) \dots (x - x_{n-1})}^{n \text{ fattori}}$$

- Chiamiamo la costante a_n :

$$p_n(x) = p_{n-1}(x) + a_n \omega_n(x)$$

Base di Newton: $\omega_0(x) \equiv 1$

$$\omega_1(x) = x - x_0$$

$$\omega_2(x) = (x - x_0)(x - x_1)$$

$$\vdots$$

$$\omega_n(x) = \prod_{k=0}^{n-1} (x - x_k)$$



$$p_n(x) = p_{n-1}(x) + a_n \omega_n(x)$$

Ragionando da una prospettiva diversa: partiamo da $n = 0$ (un solo nodo), e aggiungiamo i nodi uno per volta:

$$p_0(x) = a_0 \quad (= f(x_0))$$

$$p_1(x) = p_0(x) + a_1 \omega_1(x)$$

$$p_2(x) = p_1(x) + a_2 \omega_2(x) = p_0(x) + a_1 \omega_1(x) + a_2 \omega_2(x)$$

\vdots

$$p_n(x) = a_0 \omega_0(x) + \dots + a_n \omega_n(x)$$

Calcolo dei coefficienti a_k ?



Differenze divise

Definizione (Differenza divisa di ordine 0 relativa al nodo x_k)

$$f[x_k] := f(x_k), \quad k = 0, \dots, n$$

Definizione (Differenza divisa di ordine 1 relativa ai nodi x_k, x_{k+1})

$$f[x_0, x_1] := \frac{f(x_1) - f(x_0)}{x_1 - x_0} = \frac{f[x_1] - f[x_0]}{x_1 - x_0}$$

$$f[x_k, x_{k+1}] := \frac{f(x_{k+1}) - f(x_k)}{x_{k+1} - x_k} = \frac{f[x_{k+1}] - f[x_k]}{x_{k+1} - x_k}$$

Definizione (Differenza divisa di ordine 2 relativa ai nodi x_k, x_{k+1}, x_{k+2})

$$f[x_0, x_1, x_2] := (f[x_1, x_2] - f[x_0, x_1]) / (x_2 - x_0)$$

$$f[x_k, x_{k+1}, x_{k+2}] := (f[x_{k+1}, x_{k+2}] - f[x_k, x_{k+1}]) / (x_{k+2} - x_k)$$

Proprietà

$$a_k = f[x_0, x_1, \dots, x_k]$$

- Caso $n = 0$, polinomio di grado 0 passante per $(x_0, f(x_0))$:

$$p_0(x) = f(x_0) = f[x_0] \cdot 1 = f[x_0]\omega_0(x)$$

- Caso $n = 1$, polinomio di grado 1 passante per $(x_0, f(x_0))$ e $(x_1, f(x_1))$:

$$p_1(x) = f(x_0) + \frac{f(x_1) - f(x_0)}{x_1 - x_0}(x - x_0) = f[x_0]\omega_0(x) + f[x_0, x_1]\omega_1(x)$$



Tavola delle differenze divise

Esempio:

Siano $(x_0, y_0) = (-2, 2)$, $(x_1, y_1) = (1, -7)$, $(x_2, y_2) = (3, -5)$,
 $(x_3, y_3) = (4, -7)$

x_k	$f[x_k]$	$f[x_k, x_{k+1}]$	$f[x_k, x_{k+1}, x_{k+2}]$	$f[x_k, x_{k+1}, x_{k+2}, x_{k+3}]$
-2	2			
1	-7	$\frac{-7-2}{1+2} = -3$		
3	-5	$\frac{-5+7}{3-1} = 1$	$\frac{1+3}{3+2} = \frac{4}{5}$	
4	-7	$\frac{-7+5}{4-3} = -2$	$\frac{-2-1}{4-1} = -1$	$\frac{-1-\frac{4}{5}}{4+2} = -\frac{3}{10}$

$$p_3(x) = 2 - 3(x+2) + \frac{4}{5}(x+2)(x-1) - \frac{3}{10}(x+2)(x-1)(x-3)$$



Esempio:

E se aggiungiamo il nodo $(x_4, y_4) = (0, 2)$?

x_k	$f[x_k]$	$f[x_k, x_{k+1}]$	$f[x_k, \dots, x_{k+2}]$	$f[x_k, \dots, x_{k+3}]$	$f[x_k, \dots, x_{k+4}]$
-2	2				
1	-7	$\frac{-7-2}{1+2} = -3$			
3	-5	$\frac{-5+7}{3-1} = 1$	$\frac{1+3}{3+2} = \frac{4}{5}$		
4	-7	$\frac{-7+5}{4-3} = -2$	$\frac{-2-1}{4-1} = -1$	$\frac{-1-\frac{4}{5}}{4+2} = -\frac{3}{10}$	
0	2	$\frac{2+7}{0-4} = -\frac{9}{4}$	$\frac{-\frac{9}{4}+2}{0-3} = \frac{1}{12}$	$\frac{\frac{1}{12}+1}{0-1} = -\frac{13}{12}$	$\frac{-\frac{13}{12}+\frac{3}{10}}{0+2} = -\frac{47}{120}$

$$\begin{aligned}
 p_4(x) &= p_3(x) - \frac{47}{120} \omega_4(x) = 2 - 3(x+2) + \frac{4}{5}(x+2)(x-1) \\
 &\quad - \frac{3}{10}(x+2)(x-1)(x-3) - \frac{47}{120}(x+2)(x-1)(x-3)(x-4)
 \end{aligned}$$



Proprietà delle differenze divise

Proprietà

Le differenze divise non dipendono dall'ordine dei nodi

$$f[x_0, \dots, x_k] = f[P(x_0, \dots, x_k)]$$

dove $P(x_0, \dots, x_k)$ è una qualunque permutazione dei nodi.

Osservazione

Se i nodi sono distinti, i denominatori delle differenze divise sono sempre diversi da zero \rightsquigarrow tavola ben definita

Se i nodi sono coincidenti, si può comunque dare un significato alla differenza divisa:

$$f[\underbrace{x_i, \dots, x_i}_{k+1}] := \frac{f^{(k)}(x_i)}{k!}$$



Legame tra differenze divise e derivate

Teorema

Sia $f \in C^k[a, b]$ e siano i nodi $x_i \in [a, b]$ non necessariamente distinti. Esiste ξ , con $\min x_i \leq \xi \leq \max x_i$, t.c.

$$f[x_0, x_1, \dots, x_k] = \frac{f^{(k)}(\xi)}{k!}$$

Osservazione

Conseguenza immediata è che se f è un polinomio di grado N e $k > N$,

$$f[x_0, \dots, x_k] = 0$$



Legame tra forma di Newton e polinomio di Taylor

Proprietà

$$P_n(x) = f(x_0) + (x - x_0)f[x_0, x_1] + (x - x_0)(x - x_1)f[x_0, x_1, x_2] + \dots (x - x_0)(x - x_1) \dots (x - x_{n-1})f[x_0, x_1, \dots, x_n]$$

Immaginiamo che tutti i nodi “collassino” su uno solo:

$$P_n(x) = f(x_0) + (x - x_0)f[x_0, x_0] + (x - x_0)^2f[x_0, x_0, x_0] + \dots (x - x_0)^nf[x_0, x_0, \dots, x_0]$$

Ricordando il significato delle differenze divise con nodi coincidenti ($f[\underbrace{x_0, \dots, x_0}_{k+1}] := \frac{f^{(k)}(x_0)}{k!}$), posso vedere il polinomio di Taylor come “caso particolare” del polinomio di Newton con nodi coincidenti!



Interpolazione polinomiale a tratti (non raccordata)

Interpolazione polinomiale: infittire nodi non garantisce miglior comportamento di $p_n(x)$

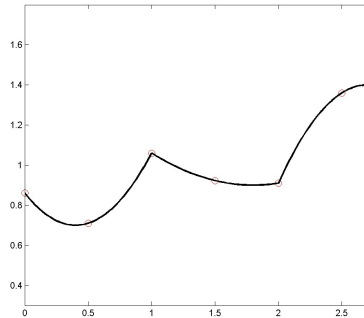
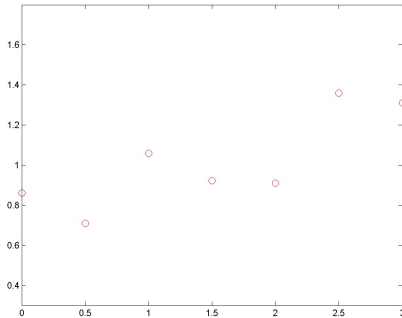
Interpolazione polinomiale a tratti:

- 1 fissiamo a priori il grado r (piccolo) del polinomio
- 2 partizioniamo $[a, b]$ in tanti intervallini tramite $n + 1$ nodi x_i :

$$a = x_0 < x_1 < \dots < x_n = b$$

- 3 ogni $r + 1$ nodi, costruiamo un polinomio diverso





- 1 infittire i nodi non significa aumentare r ma il numero di tratti di polinomio
- 2 se f è regolare converge uniformemente



Interpolazione tramite spline

Definizione (spline)

Siano x_i nodi che formano una partizione di $[a, b]$:

$$a \equiv x_0 < x_1 < \dots < x_{n-1} = x_n \equiv b$$

Fissato $d \geq 1$, $s_d(x)$ è una funzione spline relativa alla partizione di $[a, b]$ se:

- ❶ $s_d(x)$ è un polinomio di grado (al più) d su ogni $[x_i, x_{i+1}]$
- ❷ $s_d^{(k)}(x)$ è una funzione continua in $[a, b] \forall k = 0, \dots, d-1$



Definizione (spline interpolante)

Dati (x_i, y_i) , $i = 0, \dots, n$, $s_d(x)$ è una spline interpolante i dati se:

- ① *$s_d(x)$ è una funzione spline relativa alla partizione $\{x_i\}$*
- ② *$s_d(x_i) = y_i$, $i = 0, \dots, n$*



Gradi di libertà

- ① numero di parametri: $(d + 1) \times n = dn + n$
- ② numero di condizioni:
 - ① $n + 1$ condizioni di interpolazione
 - ② $(n - 1) \times d$ condizioni di raccordo nei nodi interni
 - ③ Totale: $n + 1 + d \times (n - 1) = n + 1 + dn - d$
- ③ Differenza: $d - 1$
- ④ Nel caso più utilizzato ($d = 3$, spline cubiche) mancano 2 condizioni

Le spline cubiche vengono dette...

- ① **naturali**: $s_3''(x_0) = 0, s_3''(x_n) = 0$
- ② **periodiche**: $s_3'(x_0) = s_3'(x_n), s_3''(x_0) = s_3''(x_n)$
- ③ **vincolate**: $s_3'(x_0) = y_0', s_3'(x_n) = y_n'$, con y_0', y_n' dati
- ④ **not-a-knot**: $s_3'''(x)$ continua anche in x_1 e x_{n-1}



Convergenza delle spline cubiche

Teorema

Sia $s_3(x)$ la spline cubica interpolante i dati (x_i, y_i) per $i = 0, \dots, n$ con condizioni di tipo 1,2,3. Sia $h = \max_i h_i$, $h_i = x_i - x_{i-1}$. Se $f \in C^2[a, b]$ per $h \rightarrow 0$ si ha

$$\|s_3^{(p)} - f^{(p)}\|_\infty = o(h^{2-p}), \quad p = 0, 1, 2$$

Se $f \in C^k[a, b]$, $k = 3, 4$ e $\frac{h}{h_i} \leq \text{cost.}$ per $h \rightarrow 0$ si ha

$$\|s_3^{(p)} - f^{(p)}\|_\infty = \begin{cases} o(h^{3-p}) & k = 3 \\ O(h^{4-p}) & k = 4 \end{cases} \quad p = 0, 1, 2, 3$$

Per regolarità di f maggiori la velocità di convergenza non migliora.



Proprietà delle spline cubiche

Teorema

Tra tutte le funzioni $f \in C^2[a, b]$ che assumono valori y_i nei nodi x_i e soddisfacenti condizioni di tipo 1,2,3, le spline cubiche sono le sole funzioni che minimizzano l'integrale

$$C[f] = \int_{x_0}^{x_n} [f''(x)]^2 dx$$

Le spline naturali godono di una proprietà di minimo assoluto.

$$\frac{f''(x)}{(1 + f'(x)^2)^{3/2}} = \text{curvatura di } f \text{ nel punto } x$$

\Rightarrow se $|f'(x)|$ è sufficientemente piccolo $C[f]$ è una approssimazione della curvatura globale di f in $[a, b]$.



Comandi MATLAB

- ① `a = polyfit(x,y,n)`
`yy = polyval(a,xx)`
- ② `yy = spline(x,y,xx)`

Esercizio proposto

Scrivere uno script per interpolare i dati relativi all'Ossido di Azoto con una spline cubica usando il comando `spline`



Metodo dei minimi quadrati

Limiti dell'interpolazione:

- interpolazione polinomiale globale non adatta per n grandi
- interpolazione in generale non adatta per dati sperimentali (molti dati + affetti da errore)



Fissiamo a priori un **modello** per i nostri dati in uno spazio di dimensione m (tipicamente $m \ll n$): se $\phi_i(x)$, $i = 1, \dots, m$ sono m funzioni linearmente indipendenti cerchiamo

$$f_m(x) = c_1\phi_1(x) + c_2\phi_2(x) + \dots + c_m\phi_m(x)$$

Esempi:

- $f_2(x) = c_1 + c_2x$
- $f_3(x) = c_1 + c_2x + c_3x^2$
- $f_4(x) = c_1 + c_2x + c_3 \cos(x) + c_4 \sin(x)$
- $f_2(x) = c_1 + c_2 \exp(x)$
- ...



Come si determinano i coefficienti c_k ?

$$f_m(x_i) = y_i \quad NO$$

$$\min_{c \in \mathbb{R}^m} \sum_{i=1}^n (f_m(x_i) - y_i)^2$$

$$\min_{c \in \mathbb{R}^m} \sum_{i=1}^n \left(\sum_{k=1}^m c_k \phi_k(x_i) - y_i \right)^2$$

Definizione

$f_m(x)$ è la **migliore approssimazione nel senso dei minimi quadrati**



Notazione matriciale: posto

$$A = \begin{pmatrix} \phi_1(x_1) & \phi_2(x_1) & \dots & \phi_m(x_1) \\ \phi_1(x_2) & \phi_2(x_2) & \dots & \phi_m(x_2) \\ \vdots & \vdots & \ddots & \vdots \\ \phi_1(x_n) & \phi_2(x_n) & \dots & \phi_m(x_n) \end{pmatrix} \in \mathbb{R}^{n \times m}$$

$$y = (y_1, \dots, y_n)^T \in \mathbb{R}^n, \quad c = (c_1, \dots, c_m)^T \in \mathbb{R}^m$$

$$\min_{c \in \mathbb{R}^m} \sum_{i=1}^n \left(\underbrace{\sum_{k=1}^m c_k \phi_k(x_i)}_{Ac} - y_i \right)^2 = \min_{c \in \mathbb{R}^m} \|Ac - y\|_2^2$$

Quindi si tratta di risolvere il sistema lineare $Ac = y$ nel senso dei minimi quadrati.



$\|Ac - y\|_2^2$ funzionale quadratico convesso: minimo assunto in corrispondenza degli zeri del gradiente, quindi per c che soddisfa

$$\nabla \|Ac - y\|_2^2 = A^T Ac - A^T y = 0$$

$A^T Ac = A^T y$ è detto sistema delle **equazioni normali**

Proprietà

$A^T A$ simmetrica semidefinita positiva. Se colonne di A linearmente indipendenti, $A^T A$ simmetrica **definita** positiva. Se $\phi_k(x)$ linearmente indipendenti e nodi distinti, sicuramente colonne di A linearmente indipendenti.



Svantaggi dell'approccio precedente:

- costoso costruire $A^T A$
- tipicamente mal condizionata

Alternativa **computazionale**: uso della fattorizzazione QR (cfr. lezioni del Prof. Canuto su soluzione di sistemi lineari nel senso dei minimi quadrati)



Comandi MATLAB

- 1 `polyfit` se $\phi_k(x)$ polinomiali
- 2 `fit`



Esercizio proposto

Approssimare i dati dell'esempio sull'Ossido di Azoto nel senso dei minimi quadrati con le seguenti funzioni modello:

① $f_m(x) = c_0 + c_1x$

② $f_m(x) = c_0 + c_1x + c_2x^2$

③ $f_m(x) = c_0 + c_1x + c_2 \sin\left(\frac{2\pi}{20}x\right) + c_3 \cos\left(\frac{2\pi}{20}x\right)$

rappresentando il grafico sovrapposto ai dati