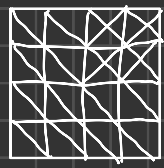
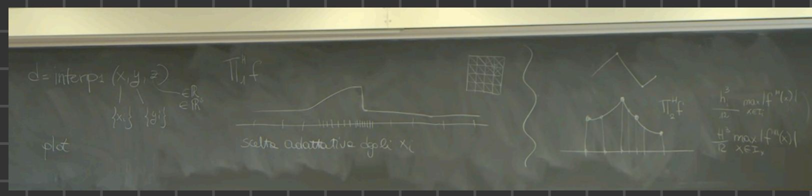
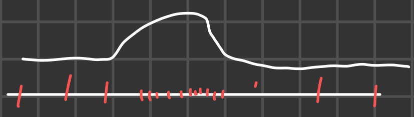
Lezione 13

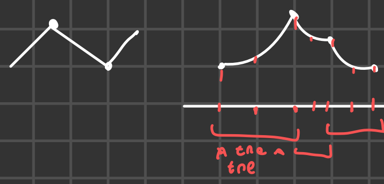




- interp. lineare a tratti
  - matlab: interp1 (x, y, z) ci sa $\Pi_1^H f$ — plot usa interp1
    - x: vettore dei nodi
    - y: vettore dei dati f
    - z: numero reale o insieme nello quale vogliamo valutare
  - matlab fa una scelta adattativa dei nodi (che si adatta alla funzione)



- se invece



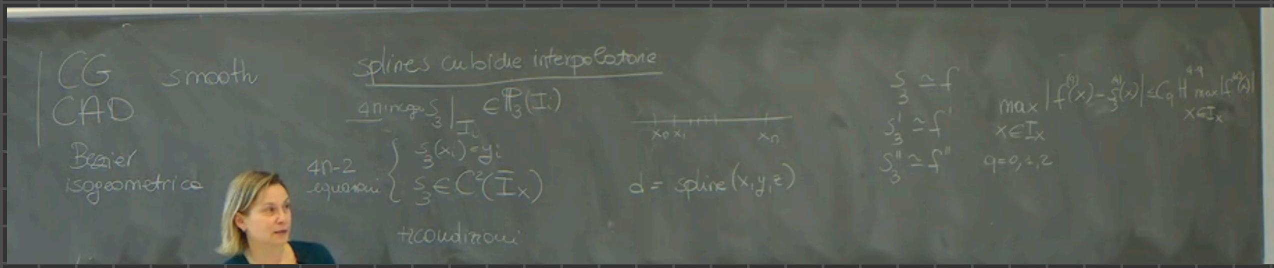
$\Pi_2^H f$ — a tre a tre

che vantaggio da? $\frac{h_i^3}{n} \max_{x \in I_i} |f'''(x)| \sim \frac{H^3}{12} \max_{x \in I_x} |f'''(x)|$ approx quadratica l'accuratezza migliora

Problema? derivata terza (richiesta di regolarità maggiore) potrebbe avere delle spunie (oscillazioni)

- nell'ambiente CGI e CAD ho bisogno di più regolarità $\sim$ smooth (cadrebbe linee spezzate) — splines cubiche interpolatorie (unisco tratti di cubica) $\sim S_3|_{I_i} \in \mathbb{P}_3(I_i)$, $S_3(x_i) = y_i$ (stiamo aumentando localmente il grado) $\sim S_3 \in C^2(\bar{I}_x)$ (globalmente)

- in alone Bezier isogeometrico

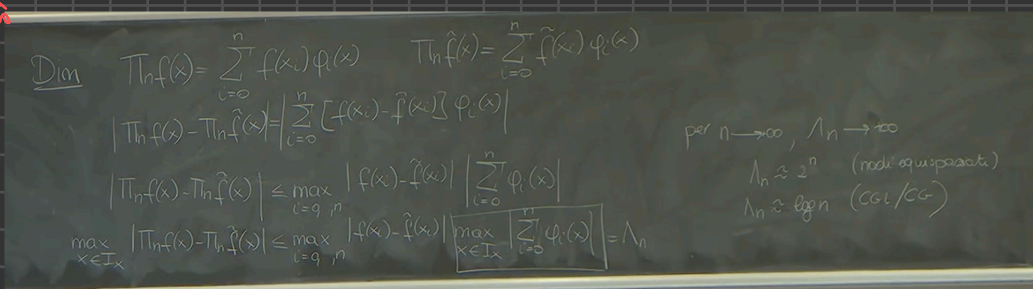


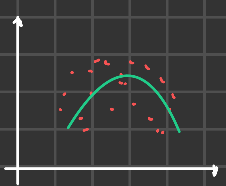
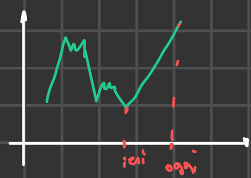
- Lagrange
  - supponiamo di avere le coppie di dati da interpolare $\{(x_i, f(x_i))\}_{i=0}^n$ x_i distinti $\Rightarrow \Pi_n f$; dati perturbati $\{(x_i, \tilde{f}(x_i))\}_{i=0}^n \Rightarrow \Pi_n \tilde{f}$
  - com'è legata $f(x_i) - \tilde{f}(x_i)$ perturbazione sui dati, $\Pi_n f(x) - \Pi_n \tilde{f}(x)$ perturbazione della soluzione
  - $\max_{x \in I_x} |\Pi_n f(x) - \Pi_n \tilde{f}(x)| = \Lambda_n \max_{i=0,\dots,n} |f(x_i) - \tilde{f}(x_i)|$ con $\Lambda_n \equiv$ costante di Lebesgue $= \max_{x \in I_x} \left|\sum_{i=0}^n \varphi_i(x)\right|$ (polinomi caratteristici)

- Dim
  - $\Pi_n f(x) = \sum_{i=0}^n f(x_i)\varphi_i(x)$, $\Pi_n \tilde{f}(x) = \sum_{i=0}^n \tilde{f}(x_i)\varphi_i(x)$
  - $\Pi_n f(x) - \Pi_n \tilde{f}(x) = \sum_{i=0}^n [f(x_i) - \tilde{f}(x_i)]\varphi_i(x)$ $\sim$ passo al val. abs $\sim$ $|\Pi_n f(x) - \Pi_n \tilde{f}(x)| = \left|\sum_{i=0}^n [f(x_i) - \tilde{f}(x_i)]\varphi_i(x)\right|$

  $\sim |\Pi_n f(x) - \Pi_n \tilde{f}(x)| \leq \max_{i=0,\dots,n} |f(x_i) - \tilde{f}(x_i)| \left|\sum_{i=0}^n \varphi_i(x)\right| \sim \max_{x \in I_x} |\Pi_n f(x) - \Pi_n \tilde{f}(x)| \leq \max_{i=0,\dots,n} |f(x_i) - \tilde{f}(x_i)| \max_{x \in I_x} \left|\sum_{i=0}^n \varphi_i(x)\right| = \Lambda_n$

  - per $n \to \infty$, $\Lambda_n \to \infty$ e $\Lambda_n \simeq 2^n$ (nodi equispaziati) (potrei avere Runge e mi chiede una regolarità)
    $\Lambda_n \simeq \log n$ (nodi CGL/GG)



- Approssimazione minimi quadrati
  -  estrapolazione (ricostruisco un andamento e cerco di prevedere ciò che può succedere) 

  ieri oggi

  - approx ai minimi quadrati: $\{(x_i, y_i)\}_{i=0}^n$, $\tilde{f} \in \mathbb{P}_m$ (m può essere libero rispetto ai dati, generalmente $m < n$) e in più è tale che $\sum_{i=0}^n [y_i - \tilde{f}(x_i)]^2 \leq \sum_{i=0}^n [y_i - p_m(x_i)]^2 \quad \forall p_m \in \mathbb{P}_m$ [def ai minimi quadrati]
  - non è detto che esista, perdo $\tilde{f}(x_i) = y_i$
  - quando $m = n$ coincide con quello di interpolazione $\tilde{f}(x) = \Pi_n f(x)$, giustifica il fatto che polyfit chieda anche n
  - $\boxed{\text{ess}}$ $m=1$ si trova la retta di regressione
  - $\tilde{f}(x) = a_0 + a_1 x + \dots + a_m x^m$ che si vanno a confrontare $p_m(x) = b_0 + b_1 x + \dots + b_m x^m$
  - sia $\Phi(b_0, \dots, b_m) = \sum_{i=0}^n [y_i - p_m(x_i)]^2$ e $\Phi(a_0, \dots, a_m) = \sum_{i=0}^n [y_i - \tilde{f}(x_i)]^2 \sim \Phi(a_0, \dots, a_m) \leq \Phi(b_0, \dots, b_m) \quad \forall (b_0, \dots, b_m) \in \mathbb{R}$
  - $\sim \Phi(a_0, \dots, a_m) = \min \Phi(b_0, \dots, b_m)$

• per $n=1$ retta tangente

$$\hat{f}(x) = a_0 + a_1 x \qquad \exists!\ a_0, a_1 \in \mathbb{R}$$

$$p_1(x) = b_0 + b_1 x$$

$$\sim \quad \Phi(b_0, b_1) = \sum_{i=0}^{n} [y_i - \underbrace{b_0 - b_1 x_i}_{p_1(x_i)}]^2 = \sum_{i=0}^{n} y_i^2 + b_0^2 + b_1^2 x_i^2 - 2 y_i b_0 - 2 y_i b_1 x_i + 2 b_0 b_1 x_i \quad \sim$$

$$\frac{\partial \Phi}{\partial b_0}(b_0, b_1) = \sum_{i=0}^{n} [2b_0 - 2y_i + 2b_1 x_i]$$

$$\frac{\partial \Phi}{\partial b_1}(b_0, b_1) = \sum_{i=0}^{n} [2b_1 x_i^2 - 2y_i x_i + 2 b_0 x_i]$$

sto prendendo $\partial(b_0 b_1)$ e sto imponendo zero

ne consegue

unico termine che non dipende dalle incognite

$$\sum_{i=0}^{n} [2a_0 - 2y_i + 2a_1 x_i] = 0$$

$$\sum_{i=0}^{n} [2a_1 x_i^2 - 2y_i x_i + 2a_0 x_i] = 0$$

$$\sim \quad \begin{bmatrix} B_{11} & B_{12} \\ B_{21} & B_{22} \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \end{bmatrix} = \begin{bmatrix} \ell_0 \\ \ell_1 \end{bmatrix}$$

$$\underbrace{\begin{bmatrix} \sum_{i=0}^{n} 1 & \sum_{i=0}^{n} x_i \\ \sum_{i=0}^{n} x_i & \sum_{i=0}^{n} x_i^2 \end{bmatrix}}_{\text{è SDP}} \begin{bmatrix} a_0 \\ a_1 \end{bmatrix} = \begin{bmatrix} \sum_{i=0}^{n} y_i \\ \sum_{i=0}^{n} y_i x_i \end{bmatrix}$$

• in generale con $\tilde{f} \in \mathbb{P}_m$

$$\begin{bmatrix} \sum x_i^0 & \sum x_i^1 & \cdots & \sum x_i^m \\ \sum x_i^1 & \sum x_i^2 & \cdots & \sum x_i^{m+1} \\ \vdots & & & \vdots \\ \sum x_i^m & \cdots & \cdots & \sum x_i^{2m} \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \\ \vdots \\ a_n \end{bmatrix} = \begin{bmatrix} \sum y_i \\ \sum x_i y_i \\ \vdots \\ \sum x_i^m y_i \end{bmatrix}$$

sistema delle equazioni normali