



# Approssimazione Polinomiale

## Laboratorio di Calcolo Numerico

Federico Piazzon

Email: [fpiazzon@math.unipd.it](mailto:fpiazzon@math.unipd.it)

17 Maggio 2022

# Soluzione equazioni normali con QR

Sia  $V = (\phi_j(x_i))_{i=1,2,\dots,m+1,j=1,2,\dots,n+1}$  la matrice di Vandermonde rettangolare sui nodi di approssimazione e supponiamo di aver calcolato la fattorizzazione

$$V = QR$$

I coefficienti  $c$  sulla base  $\phi_1, \dots, \phi_{n+1}$  dell'approssimazione ai minimi quadrati della funzione  $f$  sono la soluzione delle equazioni normali

$$V^t V c = V^t (f(x_1), \dots, f(x_{n+1}))^t.$$

Abbiamo visto come sia più stabile il calcolo della soluzione come soluzione (calcolabile con sost. all'indietro) di

$$R_0 c = Q_0^t (f(x_1), \dots, f(x_{n+1}))^t$$

dove  $R_0 = R(1 : n+1, :)$  e  $Q_0 = Q(:, 1 : n+1)$ .

# Valutazione polinomio approssimante

Noti i coefficienti del polinomio di approssimazione ai minimi quadrati per la sua valutazione sui punti  $x_1^{eval}, \dots, x_M^{eval}$  è sufficiente calcolare la Vandermonde di valutazione

$$V^{eval} := (\phi_j(x_i^{eval}))_{i=1,2,\dots,M, j=1,2,\dots,n+1}$$

e calcolare il prodotto matrice vettore

$$\begin{pmatrix} p(x_1^{eval}) \\ \vdots \\ p(x_m^{eval}) \end{pmatrix} = V^{eval} * c.$$

## Esercizio 1

Si crei una function `MyPolyfit.m` avente la chiamata `[peval,c]=MyPolyfit(xfit,xeval,yfit,deg)` che calcoli coefficienti dell'approssimante e valutazione della stessa sui nodi `xeval` seguendo il metodo QR. **NB:**

- usare come base dei polinomi la base di Chebyshev (presupponiamo che  $[a, b] := [-1, 1]$ ), usare a tal fine la function `chebvand1d.m` fornita dal docente,
- usare la sostituzione all'indietro per la soluzione del sistema triangolare, usare a tal fine la function `SostituzioneIndietro.m` fornita dal docente.

## Esercizio 2

Si crei uno script `rungefit.m` che per i gradi  $n = 2, 4, \dots, 150$  calcoli usando `MyPolyfit.m` l'approssimazione ai minimi quadrati della funzione di Runge  $f(x) : 1/(25x^2 + 1)$  costruita su 2000 punti equispaziati in  $[-1, 1]$  e la valuti su 10000 punti equispaziati.

Si deve calcolare il massimo errore sui punti di valutazione per ogni grado e fare un grafico semilogaritmico dell'errore.

## Esercizio 3

Si modifichi lo script precedente costruendo **anche** l'approssimante relativa a 2000 nodi di Chebyshev-Lobatto  $x_i := \cos(i\pi/n)$ ,  $i = 0, 1, \dots, n$  e includendo nel grafico anche l'errore di questa successione di approssimanti.

## Esercizio 4

Si modifichi la function `MyPolyfit.m` per ottenere `MyPolyfit_unstable.m` in cui invece che la base di Chebyshev venga usata la base canonica (sugg  $V = x_{fit}^{(0:deg)}$ ) e, invece che usare il metodo QR vengano risolte le eq. Normal con il backslash.

## Esercizio 5

Si modifichi nuovamente lo script per includere anche il calcolo delle approssimanti, costruite su punti di Chebyshev e su p.ti equispaziati, effettuato con `MyPolyfit_unstable.m`. Si aggiungano i plot degli errori alla figura.

# Risultati

